Understanding Intrusion and Correlation Data Structures

The eStreamer service transmits a number of data record types to deliver requested events and metadata to the client. This chapter describes the structures of data records for the following types of event data:

- intrusion events data and event extra data generated by managed devices
- correlation (compliance) events generated by the Management Center
- metadata records

The following sections in this chapter define the event message structures:

- Intrusion Event and Metadata Record Types, page 3-1.

For a general overview eStreamer's message format for transmitting data records, see Event Data Message Format, page 2-19.

Intrusion Event and Metadata Record Types

The table that follows lists all currently supported record types for intrusion events, intrusion event extra data, and metadata messages. The data for these record types is in fixed-length fields. By contrast, correlation event records contain one or more levels of nested data blocks with variable lengths. The table below provides a link to the chapter subsection that defines the associated data record structure.

For some record types, eStreamer supports more than one version. The table indicates the status of each version (current or legacy). A current record is the latest version. A legacy record has been superseded by a later version but can still be requested from eStreamer.

Table 3-1 Intrusion Event and General Metadata Record Types

Record Type	Block Type	Series	Description	Record Status	Data Format Described in...
2	N/A	N/A	Packet Data (Version 4.8.0.2+)	Current	Packet Record 4.8.0.2+, page 3-5
4	N/A	N/A	Priority Metadata	Current	Priority Record, page 3-6
9	20	1	Intrusion Impact Alert	Legacy	Intrusion Impact Alert Data, page B-62
9	153	1	Intrusion Impact Alert	Current	Intrusion Impact Alert Data 5.3+, page 3-18
62	N/A	2	User Metadata	Current	User Record, page 3-21

Table 3-1 Intrusion Event and General Metadata Record Types (continued)

Record Type	Block Type	Series	Description	Record Status	Data Format Described in...
66	N/A	N/A	Rule Message Metadata (Version 4.6.1+)	Current	Rule Message Record for 4.6.1+, page 3-22
67	N/A	N/A	Classification Metadata (Version 4.6.1+)	Current	Classification Record for 4.6.1+, page 3-23
69	N/A	N/A	Correlation Policy Metadata (Version 4.6.1+)	Current	Correlation Policy Record, page 3-25
70	N/A	N/A	Correlation Rule Metadata (Version 4.6.1+)	Current	Correlation Rule Record, page 3-26
104	N/A	N/A	Intrusion Event (IPv4) Record 4.9 - 4.10.x	Legacy	earlier versions of the product
105	N/A	N/A	Intrusion Event (IPv6) Record 4.9-4.10.x	Legacy	earlier versions of the product
110	4	2	Intrusion Event Extra Data (Version 4.10.0+)	Legacy	Intrusion Event Extra Data Record, page B-65
111	5	2	Intrusion Event Extra Data Metadata (Version 4.10.0+)	Legacy	Intrusion Event Extra Data Metadata, page B-66
112	128	1	Correlation Event for 5.1-5.3.x	Legacy	Correlation Event for 5.1-5.3.x, page B-336
112	156	1	Correlation Event for 5.4+	Current	Correlation Event for 5.4+, page 3-40
115	14	2	Security Zone Name Metadata	Current	Security Zone Name Record, page 3-28
116	14	2	Interface Name Metadata	Current	Interface Name Record, page 3-29
117	14	2	Access Control Policy Name Metadata	Current	Access Control Policy Name Record, page 3-30
118	15	2	Intrusion Policy Name Metadata	Current	Intrusion Policy Name Record, page 4-22
119	15	2	Access Control Rule ID Metadata	Current	Access Control Rule ID Record Metadata, page 3-31
120	N/A	N/A	Access Control Rule Action Metadata	Current	Access Control Rule Action Record Metadata, page 4-23
121	N/A	N/A	URL Category Metadata	Current	URL Category Record Metadata, page 4-24
122	N/A	N/A	URL Reputation Metadata	Current	URL Reputation Record Metadata, page 4-25
123	N/A	N/A	Managed Device Metadata	Current	Managed Device Record Metadata, page 3-33
N/A	64	2	Access Control PolicyName Data Block	Current	Access Control Policy Name Data Block, page 3-77
124	59	2	Access Control Policy Rule Reason Data Block	Current	Access Control Policy Rule Reason Data Block for 6.0+, page 3-75
125	N/A	2	Malware Event Record (Version 5.1.1+)	Current	Malware Event Record 5.1.1+, page 3-33

Table 3-1 Intrusion Event and General Metadata Record Types (continued)

Record Type	Block Type	Series	Description	Record Status	Data Format Described in...
125	24	2	Malware Event (Version 5.1.1+)	Legacy	Malware Event Data Block 5.1.1.x, page B-72
125	33	2	Malware Event (Version 5.2.x)	Legacy	Malware Event Data Block 5.2.x, page B-78
125	35	2	Malware Event (Version 5.3)	Legacy	Malware Event Data Block 5.3, page B-85
125	44	2	Malware Event (Version 5.3.1)	Legacy	Malware Event Data Block 5.3.1, page B-92
125	47	2	Malware Event (Version 5.4.x)	Legacy	Malware Event Data Block 5.4.x, page B-99
125	62	2	Malware Event (Version 6.x)	Legacy	Malware Event Data Block 6.x, page B-109
125	80	2	Malware Event (Version 7.0+)	Current	Malware Event Data Block 7.0+, page 3-89
127	14	2	Cisco Advanced Malware Protection Cloud Name Metadata (Version 5.1+)	Current	Cisco Advanced Malware Protection Cloud Name Metadata, page 3-34
128	N/A	N/A	Malware Event Type Metadata (Version 5.1+)	Current	Malware Event Type Metadata, page 3-36
129	N/A	N/A	Malware Event Subtype Metadata (Version 5.1+)	Current	Malware Event Subtype Metadata, page 3-37
130	N/A	N/A	AMP for Endpoints Detector Type Metadata (Version 5.1+)	Current	AMP for Endpoints Detector Type Metadata, page 3-37
131	N/A	N/A	AMP for Endpoints File Type Metadata (Version 5.1+)	Current	AMP for Endpoints File Type Metadata, page 3-38
132	N/A	N/A	Security Context Name	Current	Security Context Name, page 3-39
140	27	2	Rule Documentation Data Block for 5.2+	Current	Rule Documentation Data Block for 5.2+, page 3-103
207	N/A	N/A	Intrusion Event (IPv4) Record 5.0.x - 5.1	Legacy	Intrusion Event (IPv4) Record 5.0.x - 5.1, page B-2
208	N/A	N/A	Intrusion Event (IPv6) Record 5.0.x - 5.1	Legacy	Intrusion Event (IPv6) Record 5.0.x - 5.1, page B-6
260	19	2	ICMP Type Data Data Block	Current	ICMP Type Data Block, page 3-64
270	20	2	ICMP Code Data Block	Current	ICMP Code Data Block, page 3-65
282	N/A	2	Security Intelligence Category Metadata for 5.4.1+	Current	Security Intelligence Category Metadata for 5.4.1+, page 3-66
300	N/A	N/A	Realm Metadata for 6.0+	Current	Realm Metadata for 6.0+, page 3-67
301	58	2	Endpoint Profile for 6.0+	Current	Endpoint Profile Data Block for 6.0+, page 3-68
302	N/A	N/A	Security Group Metadata for 6.0+	Current	Security Group Metadata for 6.0+, page 3-69

Table 3-1 Intrusion Event and General Metadata Record Types (continued)

Record Type	Block Type	Series	Description	Record Status	Data Format Described in...
320	N/A	N/A	DNS Record Type Metadata for 6.0+	Current	DNS Record Type Metadata for 6.0+, page 3-70
321	N/A	N/A	DNS Response Type Metadata for 6.0+	Current	DNS Response Type Metadata for 6.0+, page 3-72
322	N/A	N/A	Sinkhole Metadata for 6.0+	Current	Sinkhole Metadata for 6.0+, page 3-73
350	N/A	N/A	Netmap Domain Metadata for 6.0+	Current	Netmap Domain Metadata for 6.0+, page 3-74
400	34	2	Intrusion Event Record 5.2.x	Legacy	Intrusion Event Record 5.2.x, page B-12
400	41	2	Intrusion Event Record 5.3	Legacy	Intrusion Event Record 5.3, page B-17
400	42	2	Intrusion Event Record 5.3.1	Legacy	Intrusion Event Record 5.3.1, page B-29
400	45	2	Intrusion Event Record 5.4.x	Legacy	Intrusion Event Record 5.4.x, page B-36
400	60	2	Intrusion Event Record 6.x	Legacy	Intrusion Event Record 6.x, page B-44
400	81	2	Intrusion Event Record 7.0	Legacy	Intrusion Event Record 7.0, page B-53
400	85	2	Intrusion Event Record 7.1+	Current	Intrusion Event Record 7.1+, page 3-7
500	32	2	File Event (Version 5.2.x)	Legacy	File Event for 5.2.x, page B-291
500	38	2	File Event (Version 5.3)	Legacy	File Event for 5.3, page B-295
500	43	2	File Event (Version 5.3.1)	Legacy	File Event for 5.3.1, page B-301
500	46	2	File Event (Version 5.4.x)	Current	File Event for 7.0+, page 3-79
502	32	2	File Event (Version 5.2.x)	Legacy	File Event for 5.2.x, page B-291
502	38	2	File Event (Version 5.3)	Legacy	File Event for 5.3, page B-295
502	43	2	File Event (Version 5.3.1)	Legacy	File Event for 5.3.1, page B-301
502	46	2	File Event (Version 5.4.x)	Legacy	File Event for 5.4.x, page B-307
502	56	2	File Event (Version 6.x)	Legacy	File Event for 6.x, page B-317
502	79	2	File Event (Version 7.0+)	Current	File Event for 7.0+, page 3-79
510	N/A	N/A	File Type ID Metadata for 5.3+	Current	File Type ID Metadata for 5.3+, page 3-102
511	26	2	File Event SHA Hash for 5.11-5.2.x	Legacy	File Event SHA Hash for 5.1.1-5.2.x, page B-327
511	40	2	File Event SHA Hash for 5.3+	Current	File Event SHA Hash for 5.3+, page 3-100
515	N/A	N/A	Filelog Storage Metadata for 6.0+	Current	Filelog Storage Metadata for 6.0+, page 3-107
516	N/A	N/A	Filelog Sandbox Metadata for 6.0+	Current	Filelog Sandbox Metadata for 6.0+, page 3-107
517	N/A	N/A	Filelog Spero Metadata for 6.0+	Current	Filelog Spero Metadata for 6.0+, page 3-108
518	N/A	N/A	Filelog Archive Metadata for 6.0+	Current	Filelog Archive Metadata for 6.0+, page 3-109

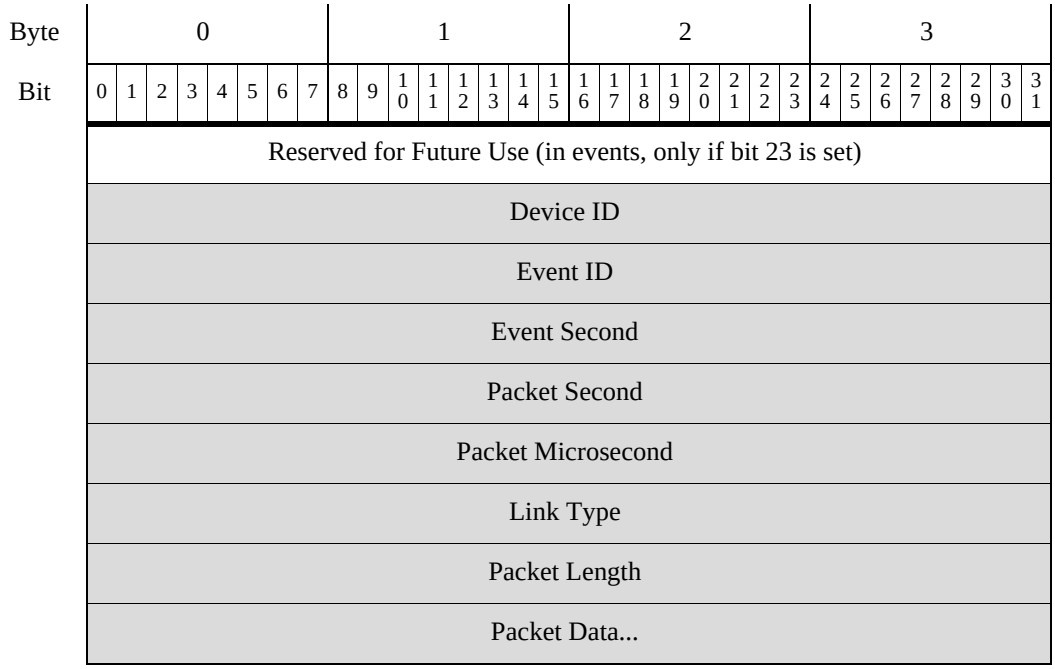
Table 3-1 Intrusion Event and General Metadata Record Types (continued)

Record Type	Block Type	Series	Description	Record Status	Data Format Described in...
519	N/A	N/A	Filelog Static Analysis Metadata for 6.0+	Current	Filelog Static Analysis Metadata for 6.0+, page 3-110
520	28	2	Geolocation Data Block for 5.2+	Current	Geolocation Data Block for 5.2+, page 3-110
530	N/A	N/A	File Policy Name for 6.0+	Current	File Policy Name for 6.0+, page 3-111
600	N/A	N/A	SSL Policy Name	Current	SSL Policy Name, page 3-113
601	51	2	SSL Rule ID	Current	SSL Rule ID, page 3-114
602	N/A	N/A	SSL Cipher Suite	Current	SSL Certificate Details Data Block for 5.4+, page 3-121
604	N/A	N/A	SSL Version	Current	SSL Version, page 3-116
605	N/A	N/A	SSL Server Certificate Status	Current	SSL Server Certificate Status, page 3-117
606	N/A	N/A	SSL Actual Action	Current	SSL Actual Action, page 3-118
607	N/A	N/A	SSL Expected Action	Current	SSL Expected Action, page 3-119
608	N/A	N/A	SSL Flow Status	Current	SSL Flow Status, page 3-119
613	N/A	N/A	SSL URL Category	Current	SSL URL Category, page 3-120
614	50	2	SSL Certificate Details Data Block for 5.4+	Current	SSL Certificate Details Data Block for 5.4+, page 3-121
700	N/A	N/A	Network Analysis Policy Record	Current	Network Analysis Policy Name Record, page 3-125

Packet Record 4.8.0.2+

The eStreamer service transmits the packet data associated with an event in a Packet record, the format of which is shown below. Packet data is sent when the Packet flag—bit 0 in the Request Flags field of a request message—is set. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record. Note that the Record Type field, which appears after the Message Length field, has a value of 2, indicating a packet record.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Header Version (1)																	Message Type (4)															
Message Length																																
Netmap ID																	Record Type (2)															
Record Length																																
eStreamer Server Timestamp (in events, only if bit 23 is set)																																



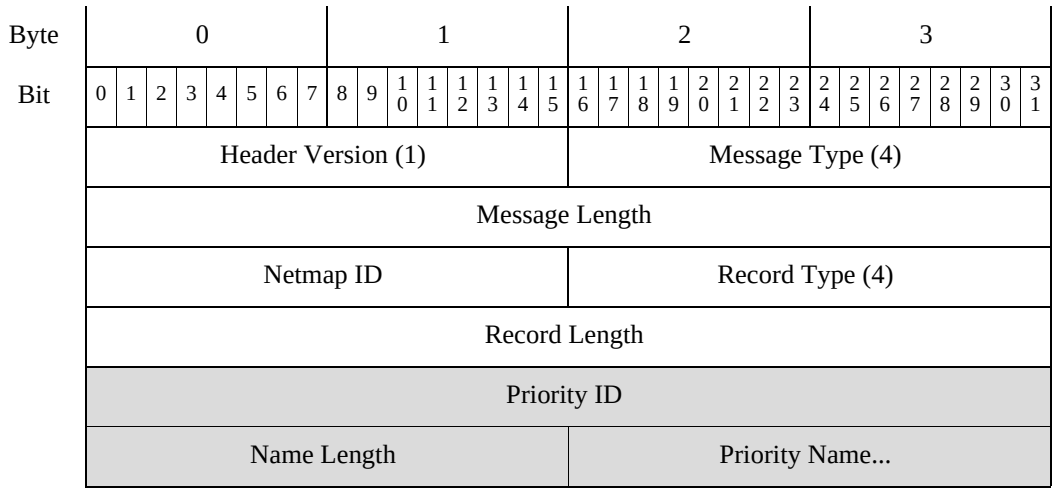
The following table describes the fields in the Packet record.

Table 3-2 Packet Record Fields

Field	Data Type	Description
Device ID	uint32	The device identification number. You can obtain device names that correlate to them by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information.
Event ID	uint32	The event identification number.
Event Second	uint32	The second (from 01/01/1970) that the event occurred.
Packet Second	uint32	The second (from 01/01/1970) that the packet was captured.
Packet Microsecond	uint32	Microsecond (one millionth of a second) increment that the packet was captured.
Link Type	uint32	Link layer type. Currently, the value will always be `1` (signifying the Ethernet layer).
Packet Length	uint32	Number of bytes included in the packet data.
Packet Data	variable	Actual captured packet data (header and payload).

Priority Record

The eStreamer service transmits the priority associated with an event in a Priority record, the format of which is shown below. (Priority information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `4`, indicating a Priority record.



The following table describes each priority-specific field.

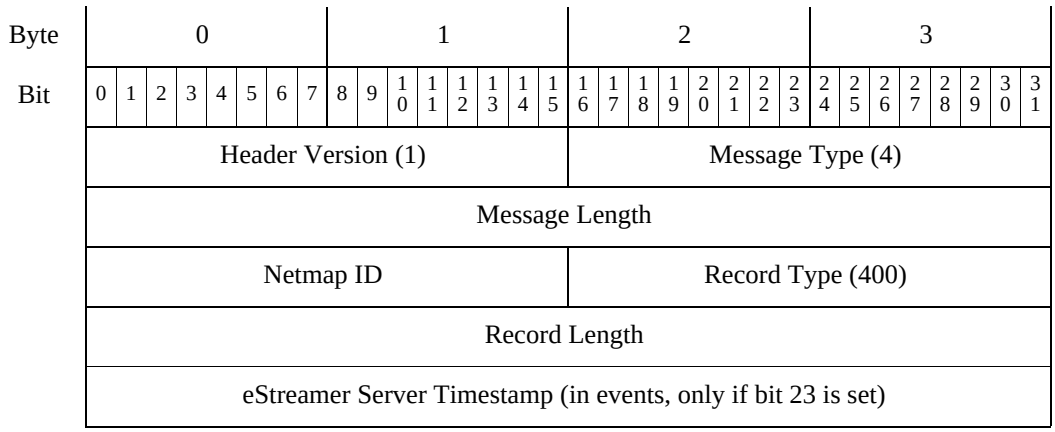
Table 3-3 Priority Record Fields

Field	Data Type	Description
Priority ID	uint32	Indicates the priority identification number.
Name Length	uint16	Number of bytes included in the priority name.
Priority Name	variable	Priority name that corresponds with the priority ID (1 - high, 2 - medium, 3 - low).

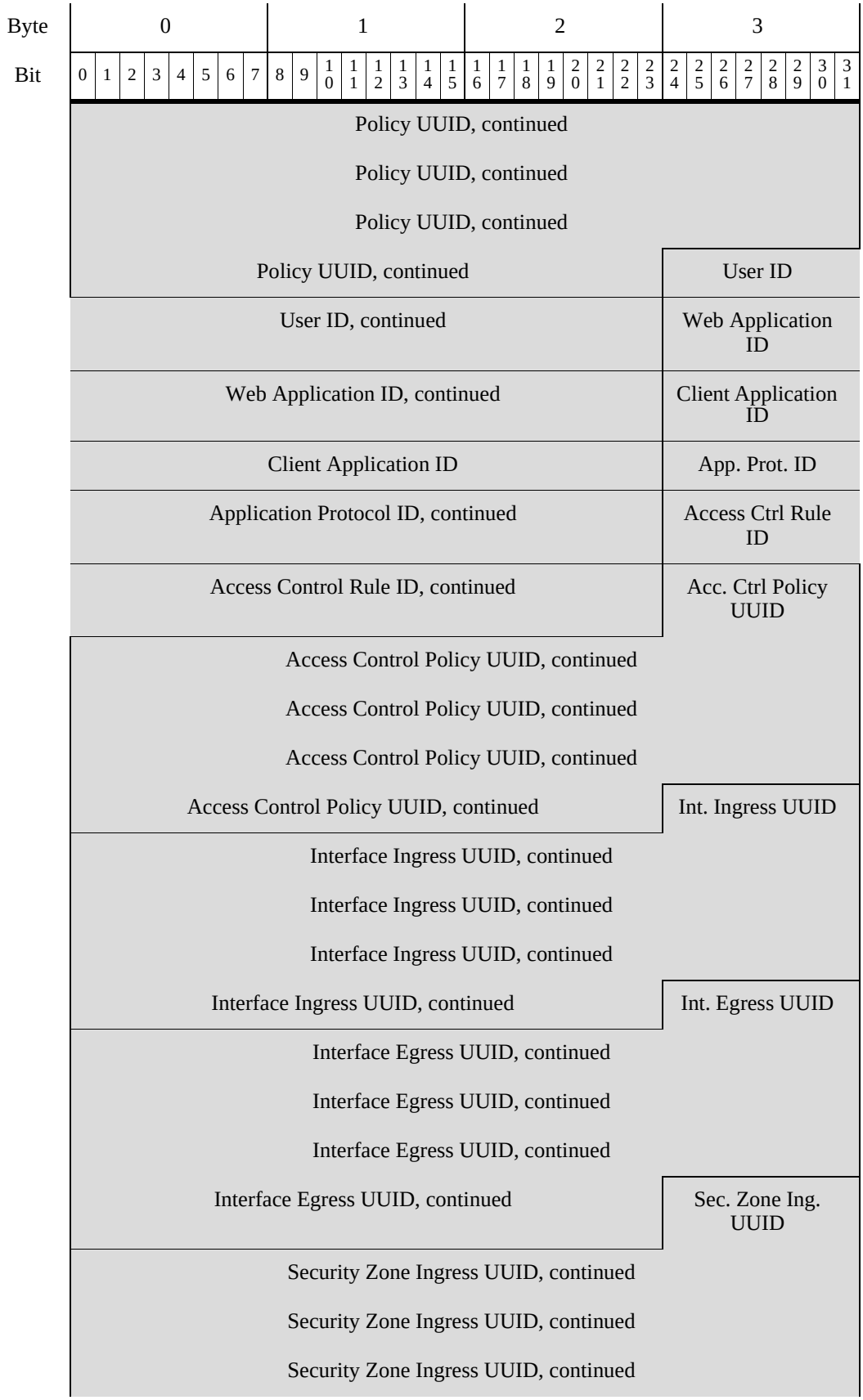
Intrusion Event Record 7.1+

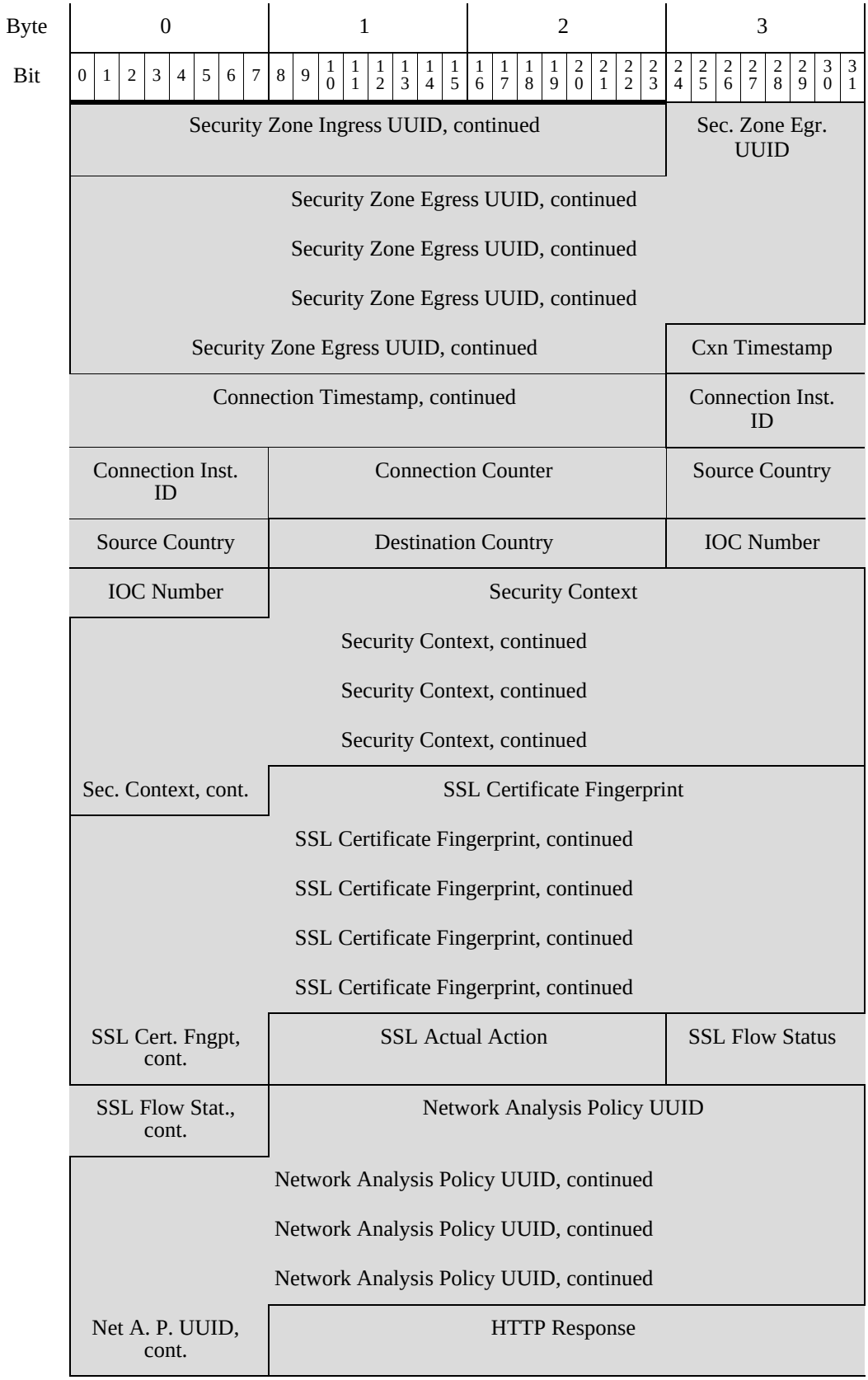
The fields in the intrusion event record are shaded in the following graphic. The record type is 400 and the block type is 85 in the series 2 set of data blocks. It supersedes block type 81. XFF fields formerly included in Extra Event Data have been added.

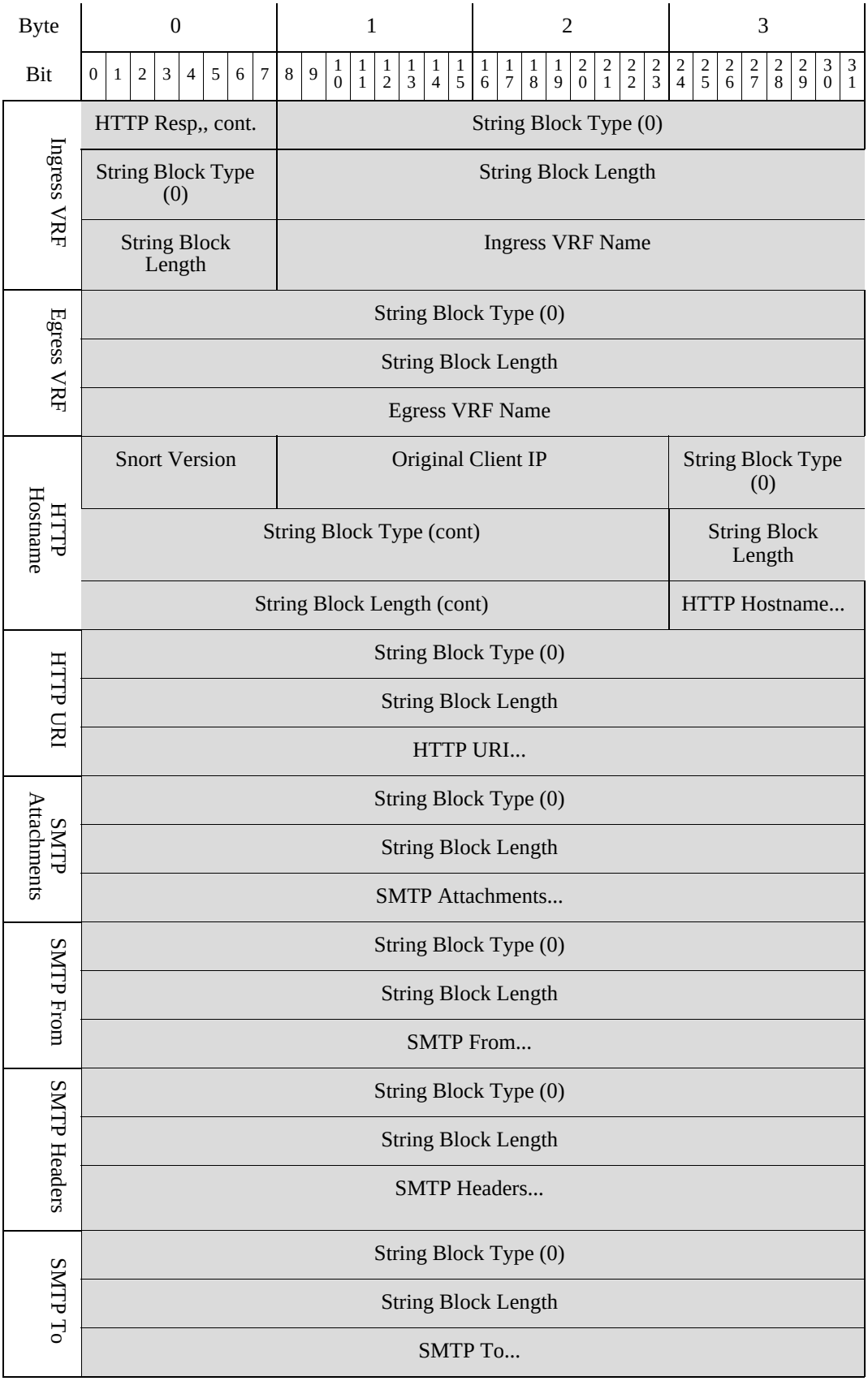
You can request 7.1+ intrusion events from eStreamer only by extended request, for which you request event type code 12 and version code 11 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests).



Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Reserved for Future Use (in events, only if bit 23 is set)																																
Block Type (85)																																
Block Length																																
Device ID																																
Event ID																																
Event Second																																
Event Microsecond																																
Rule ID (Signature ID)																																
Generator ID																																
Rule Revision																																
Classification ID																																
Priority ID																																
Source IP Address Source IP Address, continued Source IP Address, continued Source IP Address, continued																																
Destination IP Address Destination IP Address, continued Destination IP Address, continued Destination IP Address, continued																																
Source Port or ICMP Type																	Destination Port or ICMP Code														
IP Protocol ID									Impact Flags												Impact				Inline Result							
Inline Result Reason									MPLS Label																							
MPLS Label, cont.									VLAN ID																Pad							
Pad, Cont.									Policy UUID																							


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Policy UUID, continued
Policy UUID, continued
Policy UUID, continued
Policy UUID, continued
User ID
User ID, continued
Web Application ID
Web Application ID, continued
Client Application ID
Client Application ID
App. Prot. ID
Application Protocol ID, continued
Access Ctrl Rule ID
Access Control Rule ID, continued
Acc. Ctrl Policy UUID
Access Control Policy UUID, continued
Access Control Policy UUID, continued
Access Control Policy UUID, continued
Access Control Policy UUID, continued
Int. Ingress UUID
Interface Ingress UUID, continued
Interface Ingress UUID, continued
Interface Ingress UUID, continued
Interface Ingress UUID, continued
Int. Egress UUID
Interface Egress UUID, continued
Interface Egress UUID, continued
Interface Egress UUID, continued
Interface Egress UUID, continued
Sec. Zone Ing. UUID
Security Zone Ingress UUID, continued
Security Zone Ingress UUID, continued
Security Zone Ingress UUID, continued


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Security Zone Ingress UUID, continued
Sec. Zone Egr. UUID
Security Zone Egress UUID, continued
Security Zone Egress UUID, continued
Security Zone Egress UUID, continued
Security Zone Egress UUID, continued
Cxn Timestamp
Connection Timestamp, continued
Connection Inst. ID
Connection Inst. ID
Connection Counter
Source Country
Source Country
Destination Country
IOC Number
IOC Number
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
Sec. Context, cont.
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Cert. Fngpt, cont.
SSL Actual Action
SSL Flow Status
SSL Flow Stat., cont.
Network Analysis Policy UUID
Network Analysis Policy UUID, continued
Network Analysis Policy UUID, continued
Network Analysis Policy UUID, continued
Net A. P. UUID, cont.
HTTP Response


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Ingress VRF
HTTP Resp,, cont.
String Block Type (0)
String Block Type (0)
String Block Length
String Block Length
Ingress VRF Name
Egress VRF
String Block Type (0)
String Block Length
Egress VRF Name
HTTP Hostname
Snort Version
Original Client IP
String Block Type (0)
String Block Type (cont)
String Block Length
String Block Length (cont)
HTTP Hostname...
HTTP URI
String Block Type (0)
String Block Length
HTTP URI...
SMTP Attachments
String Block Type (0)
String Block Length
SMTP Attachments...
SMTP From
String Block Type (0)
String Block Length
SMTP From...
SMTP Headers
String Block Type (0)
String Block Length
SMTP Headers...
SMTP To
String Block Type (0)
String Block Length
SMTP To...

The following table describes each intrusion event record data field.

Table 3-4 Intrusion Event Record 7.1+ Fields

Field	Data Type	Description
Block Type	unint32	Initiates an Intrusion Event data block. This value is always 85.
Block Length	unint32	Total number of bytes in the Intrusion Event data block, including eight bytes for the Intrusion Event block type and length fields, plus the number of bytes of data that follows.
Device ID	unit32	Contains the identification number of the detecting managed device. You can obtain the managed device name by requesting Version 3 or 4 metadata. See Managed Device Record Metadata, page 3-33 for more information.
Event ID	uint32	Event identification number.
Event Second	uint32	UNIX timestamp (seconds since 01/01/1970) of the event's detection.
Event Microsecond	uint32	Microsecond (one millionth of a second) increment of the timestamp of the event's detection.
Rule ID (Signature ID)	uint32	Rule identification number that corresponds with the event.
Generator ID	uint32	Identification number of the Secure Firewall System preprocessor that generated the event.
Rule Revision	uint32	Rule revision number.
Classification ID	uint32	Identification number of the event classification message.
Priority ID	uint32	Identification number of the priority associated with the event.
Source IP Address	uint8[16]	Source IPv4 or IPv6 address used in the event.
Destination IP Address	uint8[16]	Destination IPv4 or IPv6 address used in the event.
Source Port or ICMP Type	uint16	The source port number if the event protocol type is TCP or UDP, or the ICMP type if the event is caused by ICMP traffic.
Destination Port or ICMP Code	uint16	The destination port number if the event protocol type is TCP or UDP, or the ICMP code if the event is caused by ICMP traffic.
IP Protocol ID	uint8	IANA-specified protocol number. For example: • `0` — IP • `1` — ICMP • `6` — TCP • `17` — UDP

Table 3-4 Intrusion Event Record 7.1+ Fields (continued)

Field	Data Type	Description
Impact Flags	bits[8]	Impact flag value of the event. The low-order eight bits indicate the impact level. Values are: • `0x01` (bit 0) — Source or destination host is in a network monitored by the system. • `0x02` (bit 1) — Source or destination host exists in the network map. • `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol. • `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event. • `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event. • `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface. • `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software. • `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only) The following impact level values map to specific priorities on the Management Center. An `x` indicates the value can be 0 or 1: • gray (0, unknown): `00X00000` • red (1, vulnerable): `XXXX1XXX`, `XXX1XXXX`, `X1XXXXXX`, `1XXXXXXX` (version 5.0+ only) • orange (2, potentially vulnerable): `00X0011X` • yellow (3, currently not vulnerable): `00X0001X` • blue (4, unknown target): `00X00001`
Impact	uint8	Impact flag value of the event. Values are: • `1` — Red (vulnerable) • `2` — Orange (potentially vulnerable) • `3` — Yellow (currently not vulnerable) • `4` — Blue (unknown target) • `5` — Gray (unknown impact)

Table 3-4 Intrusion Event Record 7.1+ Fields (continued)

Field	Data Type	Description
Inline Result	uint8	Value indicating the inline result. • 0 — Pass • 1 — Dropped • 2 — Would be dropped (but not permitted by configuration) • 3— Partially dropped • 4— Block • 5— Would block • 6— Partial block • 7— Drop • 8— Would drop • 9— Reject • 10— Would reject • 11— React • 12— Would react • 13— Rewrite • 14— Would rewrite
Inline Result Reason	uint8	Value indicating the inline result reason. • 1— Interface in Passive or Tap mode • 2— Intrusion Policy in “Detection” inspection mode • 3— Network Analysis Policy in “Detection” inspection mode • 4— Connection timed out • 5— Connection Closed (internal use) • 6— Connection Closed (internal use) • 7— Connection Closed (internal use)
MPLS Label	uint32	MPLS label.
VLAN ID	uint16	Indicates the ID of the VLAN where the packet originated.
Pad	uint16	Reserved for future use.
Policy UUID	uint8[16]	A policy ID number that acts as a unique identifier for the intrusion policy.
User ID	uint32	The internal identification number for the user, if applicable.
Web Application ID	uint32	The internal identification number for the web application, if applicable.
Client Application ID	uint32	The internal identification number for the client application, if applicable.
Application Protocol ID	uint32	The internal identification number for the application protocol, if applicable.

Table 3-4 Intrusion Event Record 7.1+ Fields (continued)

Field	Data Type	Description
Access Control Rule ID	uint32	A rule ID number that acts as a unique identifier for the access control rule.
Access Control Policy UUID	uint8[16]	A policy ID number that acts as a unique identifier for the access control policy.
Interface Ingress UUID	uint8[16]	An interface ID number that acts as a unique identifier for the ingress interface.
Interface Egress UUID	uint8[16]	An interface ID number that acts as a unique identifier for the egress interface.
Security Zone Ingress UUID	uint8[16]	A zone ID number that acts as a unique identifier for the ingress security zone.
Security Zone Egress UUID	uint8[16]	A zone ID number that acts as a unique identifier for the egress security zone.
Connection Timestamp	uint32	UNIX timestamp (seconds since 01/01/1970) of the connection event associated with the intrusion event.
Connection Instance ID	uint16	Numerical ID of the Snort instance on the managed device that generated the connection event.
Connection Counter	uint16	Value used to distinguish between connection events that happen during the same second.
Source Country	uint16	Code for the country of the source host.
Destination Country	uint 16	Code for the country of the destination host.
IOC Number	uint16	ID number of the compromise associated with this event.
Security Context	uint8[16]	ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode.
SSL Certificate Fingerprint	uint8[20]	SHA1 hash of the SSL Server certificate.
SSL Actual Action	uint16	The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include: • `0` — 'Unknown' • `1` — 'Do Not Decrypt' • `2` — 'Block' • `3` — 'Block With Reset' • `4` — 'Decrypt (Known Key)' • `5` — 'Decrypt (Replace Key)' • `6` — 'Decrypt (Resign)'

Table 3-4 Intrusion Event Record 7.1+ Fields (continued)

Field	Data Type	Description
SSL Flow Status	uint16	Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include: • 0 — 'Unknown' • 1 — 'No Match' • 2 — 'Success' • 3 — 'Uncached Session' • 4 — 'Unknown Cipher Suite' • 5 — 'Unsupported Cipher Suite' • 6 — 'Unsupported SSL Version' • 7 — 'SSL Compression Used' • 8 — 'Session Undecryptable in Passive Mode' • 9 — 'Handshake Error' • 10 — 'Decryption Error' • 11 — 'Pending Server Name Category Lookup' • 12 — 'Pending Common Name Category Lookup' • 13 — 'Internal Error' • 14 — 'Network Parameters Unavailable' • 15 — 'Invalid Server Certificate Handle' • 16 — 'Server Certificate Fingerprint Unavailable' • 17 — 'Cannot Cache Subject DN' • 18 — 'Cannot Cache Issuer DN' • 19 — 'Unknown SSL Version' • 20 — 'External Certificate List Unavailable' • 21 — 'External Certificate Fingerprint Unavailable' • 22 — 'Internal Certificate List Invalid' • 23 — 'Internal Certificate List Unavailable' • 24 — 'Internal Certificate Unavailable' • 25 — 'Internal Certificate Fingerprint Unavailable' • 26 — 'Server Certificate Validation Unavailable' • 27 — 'Server Certificate Validation Failure' • 28 — 'Invalid Action'
Network Analysis Policy UUID	uint8[16]	The UUID of the Network Analysis Policy that created the intrusion event.
HTTP Response	uint32	Response code of the HTTP Request.

Table 3-4 Intrusion Event Record 7.1+ Fields (continued)

Field	Data Type	Description
String Block Type	uint32	Initiates a String data block containing the name of the ingress VRF. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Ingress VRF name field.
Ingress VRF Name	string	The virtual router through which traffic entered the network.
String Block Type	uint32	Initiates a String data block containing the name of the egress VRF. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Egress VRF name field.
Egress VRF Name	string	The name of the virtual router through which traffic exited the network.
Snort Version	uint8	Snort version number.
Original initiator IP	uint16	Contains the IP address of the original initiator of the connection.
String Block Type	uint32	Initiates a String data block containing the names of the HTTP Hostname. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the HTTP Hostname field.
HTTP Hostname	string	Contains the host name found in the HTTP connection.
String Block Type	uint32	Initiates a String data block containing the names of the HTTP URI. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the HTTP URI field.
HTTP URI	string	Contains the Universal Resource Indicator found in the HTTP connection.
String Block Type	uint32	Initiates a String data block containing the names of the SMTP Attachments. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SMTP Attachments field.
SMTP Attachments	string	Contains the MIME attachment file name that was extracted from the MIME Content-Disposition header. For this field to be populated you must enable the SMTP preprocessor `Log MIME Attachment Names` option. Multiple attachment file names are supported.
String Block Type	uint32	Initiates a String data block containing the SMTP From address. This value is always `0`.

Table 3-4 Intrusion Event Record 7.1+ Fields (continued)

Field	Data Type	Description
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SMTP From field.
SMTP From	string	Contains the address of the email sender that was extracted from the SMTP MAIL FROM command. For this field to be populated you must enable the SMTP preprocessor `Log From Address` option. Multiple sender addresses are supported.
String Block Type	uint32	Initiates a String data block containing the SMTP Headers. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SMTP Headers field.
SMTP Headers	string	Contains the data that was extracted from the email header. To associate email headers with intrusion events for SMTP traffic you must enable the SMTP preprocessor `Log Headers` option.
String Block Type	uint32	Initiates a String data block containing the SMTP To address. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SMTP To field.
SMTP To	string	Contains the address of the email recipient that was extracted from the SMTP RCPT TO command. For this field to be populated you must enable the SMTP preprocessor `Log To Addresses` option. Multiple recipient addresses are supported.

Intrusion Impact Alert Data 5.3+

The Intrusion Impact Alert 5.3+ event contains information about impact events. It is transmitted when an intrusion event is compared to the system network map data and the impact is determined. It uses the standard record header with a record type of 9, followed by an Intrusion Impact Alert data block with a series 1 data block type of 153 in the series 1 group of blocks. (The Impact Alert data block is a type of series 1 data block. For more information about series 1 data blocks, see Understanding Discovery (Series 1) Blocks, page 4-62.)

You can request that eStreamer only transmit intrusion impact events by setting bit 5 in the Flags field of the request message. See Event Stream Request Message Format, page 2-12 for more information about request messages. Version 1 of these alerts only handles IPv4. Version 2, introduced in 5.3, handles IPv6 events in addition to IPv4.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Header Version (1)																Message Type (4)															
	Message Length																															
	Netmap ID																Record Type (9)															
	eStreamer Server Timestamp (in events, only if bit 23 is set)																															
	Reserved for Future Use (in events, only if bit 23 is set)																															
	Intrusion Impact Alert Block Type (153)																															
	Intrusion Impact Alert Block Length																															
	Event ID																															
	Device ID																															
	Event Second																															
	Impact																															
	Source IP Address Source IP Address, continued Source IP Address, continued Source IP Address, continued																															
	Destination IP Address Destination IP Address, continued Destination IP Address, continued Destination IP Address, continued																															
Impact Description	String Block Type (0)																															
	String Block Length																															
	Description...																															

The following table describes each data field in an impact event.

Table 3-5 Impact Event Data Fields

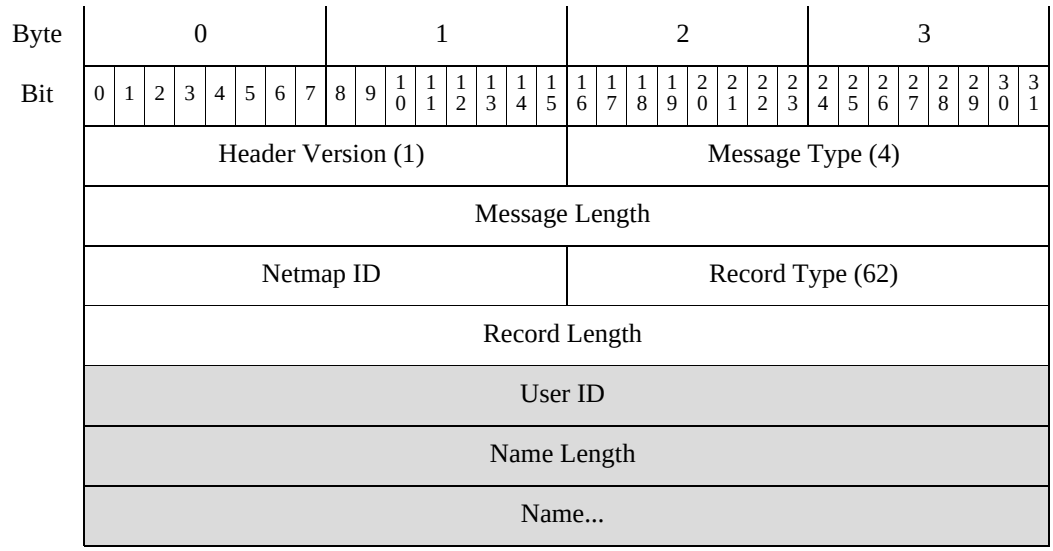
Field	Data Type	Description
Intrusion Impact Alert Block Type	uint32	Indicates that an intrusion impact alert data block follows. This field will always have a value of `153`. See Intrusion Event and Metadata Record Types, page 3-1.
Intrusion Impact Alert Block Length	uint32	Indicates the length of the intrusion impact alert data block, including all data that follows and 8 bytes for the intrusion impact alert block type and length.
Event ID	uint32	Indicates the event identification number.
Device ID	uint32	Indicates the managed device identification number.
Event Second	uint32	Indicates the second (from 01/01/1970) that the event was detected.
Impact	bits[8]	Impact flag value of the event. The low-order eight bits indicate the impact level. Values are: • `0x01` (bit 0) — Source or destination host is in a network monitored by the system. • `0x02` (bit 1) — Source or destination host exists in the network map. • `0x04` (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol. • `0x08` (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event. • `0x10` (bit 4) — There is a vulnerability mapped to the server detected in the event. • `0x20` (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface. • `0x40` (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software. • `0x80` (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only) The following impact level values map to specific priorities on the Management Center. An `x` indicates the value can be 0 or 1: • gray (0, unknown): `00X00000` • red (1, vulnerable): `XXXX1XXX`, `XXX1XXXX`, `X1XXXXXX`, `1XXXXXXX` (version 5.0+ only) • orange (2, potentially vulnerable): `00X0011X` • yellow (3, currently not vulnerable): `00X0001X` • blue (4, unknown target): `00X00001`

Table 3-5 Impact Event Data Fields (continued)

Field	Data Type	Description
Source IP Address	uint8[16]	IP address of the host associated with the impact event. This can contain either an IPv4 or IPv6 address. See IP Addresses, page 1-4 for more information.
Destination IP Address	uint8[16]	IP address of the destination IP address associated with the impact event (if applicable). This can contain either an IPv4 or IPv6 address. See IP Addresses, page 1-4 for more information. This value is 0 if there is no destination IP address.
String Block Type	uint32	Initiates a string data block that contains the impact name. This value is always set to 0. For more information about string blocks, see String Data Block, page 4-71.
String Block Length	uint32	Number of bytes in the event description string block. This includes the four bytes for the string block type, the four bytes for the string block length, and the number of bytes in the description.
Description	string	Description of the impact event.

User Record

When you request metadata, you can retrieve information about the users referenced in events generated by components in your Secure Firewall System. The eStreamer service transmits metadata containing user information for an event within a User record, the format of which is shown below. The User Record contains a user ID and the corresponding name. The user metadata record can be used to determine a user name associated with an event by correlating the metadata with the user ID value. (User information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.)



The following table describes the fields in the User record.

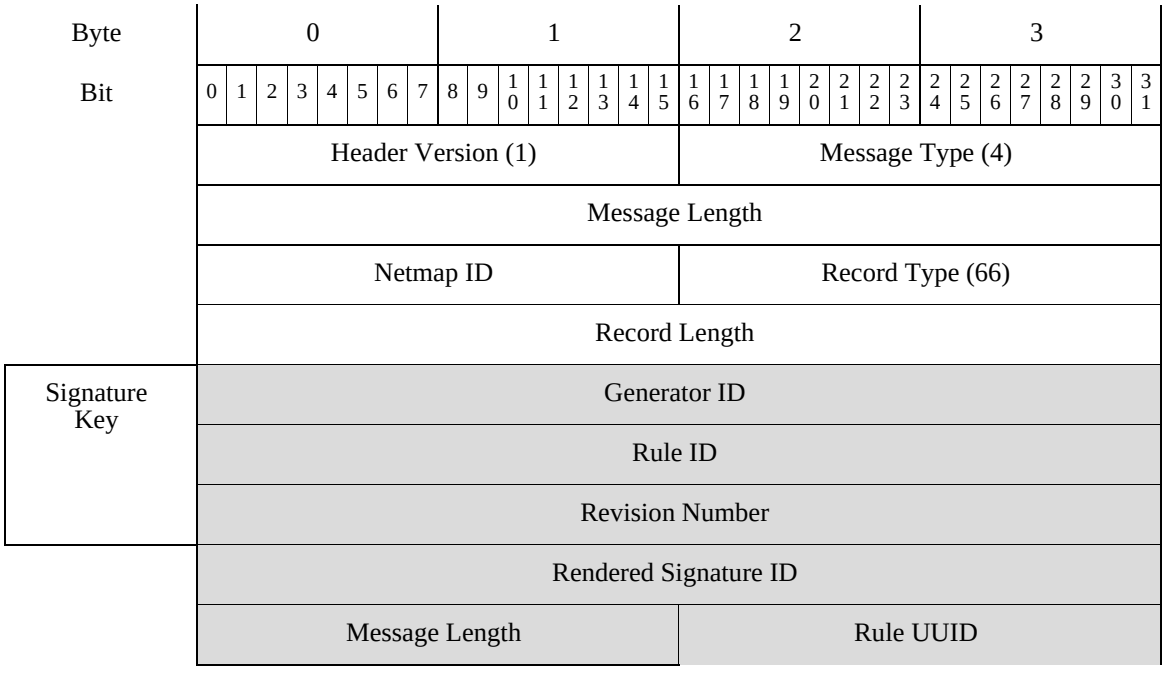
Table 3-6 User Record Fields

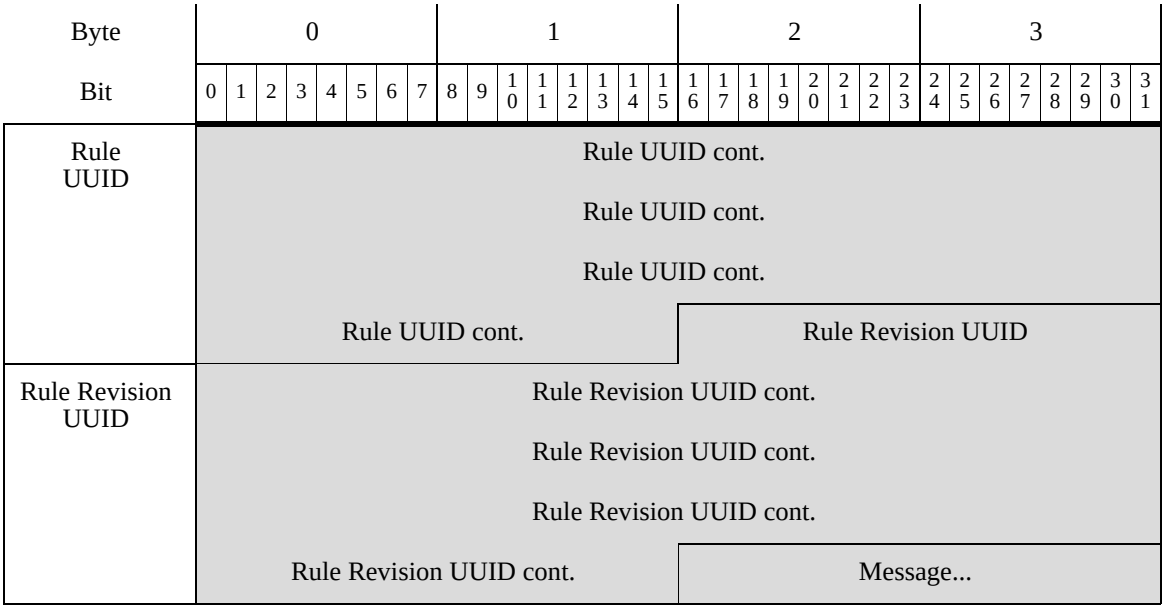
Field	Data Type	Description
User ID	uint32	The user ID number. This field is the unique key for this record.
Name Length	uint32	The number of bytes included in the user name.
Name	string	The name of the user.

Rule Message Record for 4.6.1+

Rule message information for an event is transmitted within a Rule Message record, the format of which is shown below. The eStreamer service transmits the Rule Message record for 4.6.1+ when you request Version 2 or Version 3 metadata. The Rule Message record for 4.6.1+ contains the same fields as the Rule Message record for 4.6 and lower but also has new UUID and Revision UUID fields. (Version 2, Version 3, or Version 4 metadata information is sent when the appropriate metadata flag—bit 14 for Version 2, bit 15 for Version 3, or bit 20 for Version 4 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `66`, indicating a Rule Message Version 2 record.

There are tens of thousandds of rules depending on firewall configuration. Each rule may generate an individual record rule message record. If caching metadata and requesting this record be sure to allocate sufficient memory.





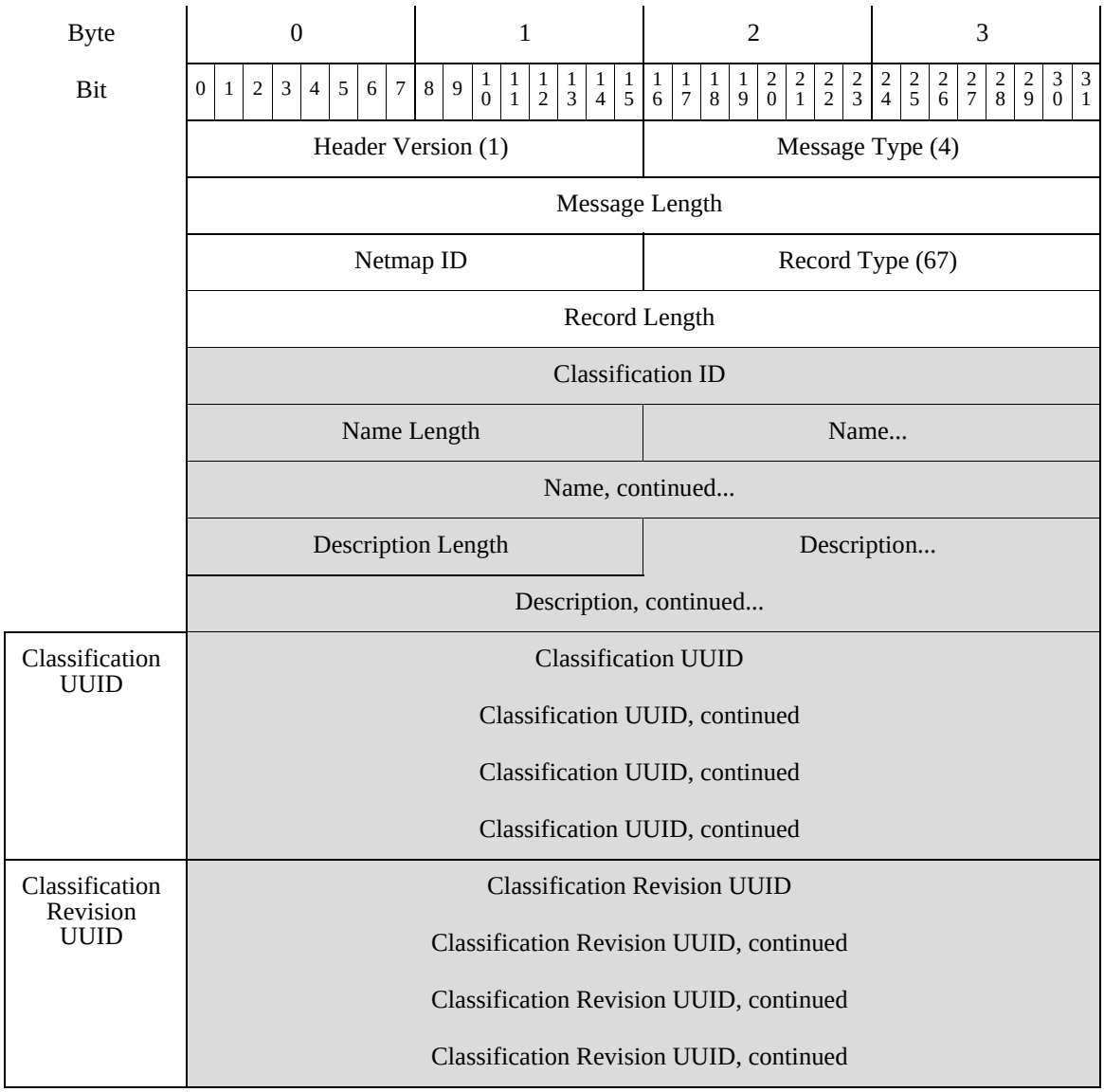
The following table describes each rule-specific field.

Table 3-7 Rule Message Record Fields

Field	Data Type	Description
Generator ID	uint32	The generator identification number.
Rule ID	uint32	The rule identification number for the local computer.
Rule Revision	uint32	The rule revision number. This is currently set to `0` for all rule messages.
Rendered Signature ID	uint32	The rule identification number rendered to the Secure Firewall System interface.
Message Length	uint16	The number of bytes included in the rule text.
UUID	uint8[16]	A rule ID number that acts as a unique identifier for the rule.
Revision UUID	uint8[16]	A rule revision ID number that acts as a unique identifier for the revision.
Message	variable	Rule message that triggered the event.

Classification Record for 4.6.1+

The eStreamer service transmits the classification information for an event in a Classification record for 4.6.1+, the format of which is shown below. The Classification record for 4.6.1+ contains the same fields as the Classification record for 4.6 and lower but also has new UUID and Revision UUID fields. (Classification information is sent when the Version 3 or Version 4 metadata flag—bit 15 or bit 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `67`, indicating a Classification Version 2 record.



The following table describes the fields in the Classification record.

Table 3-8 Classification Record Fields

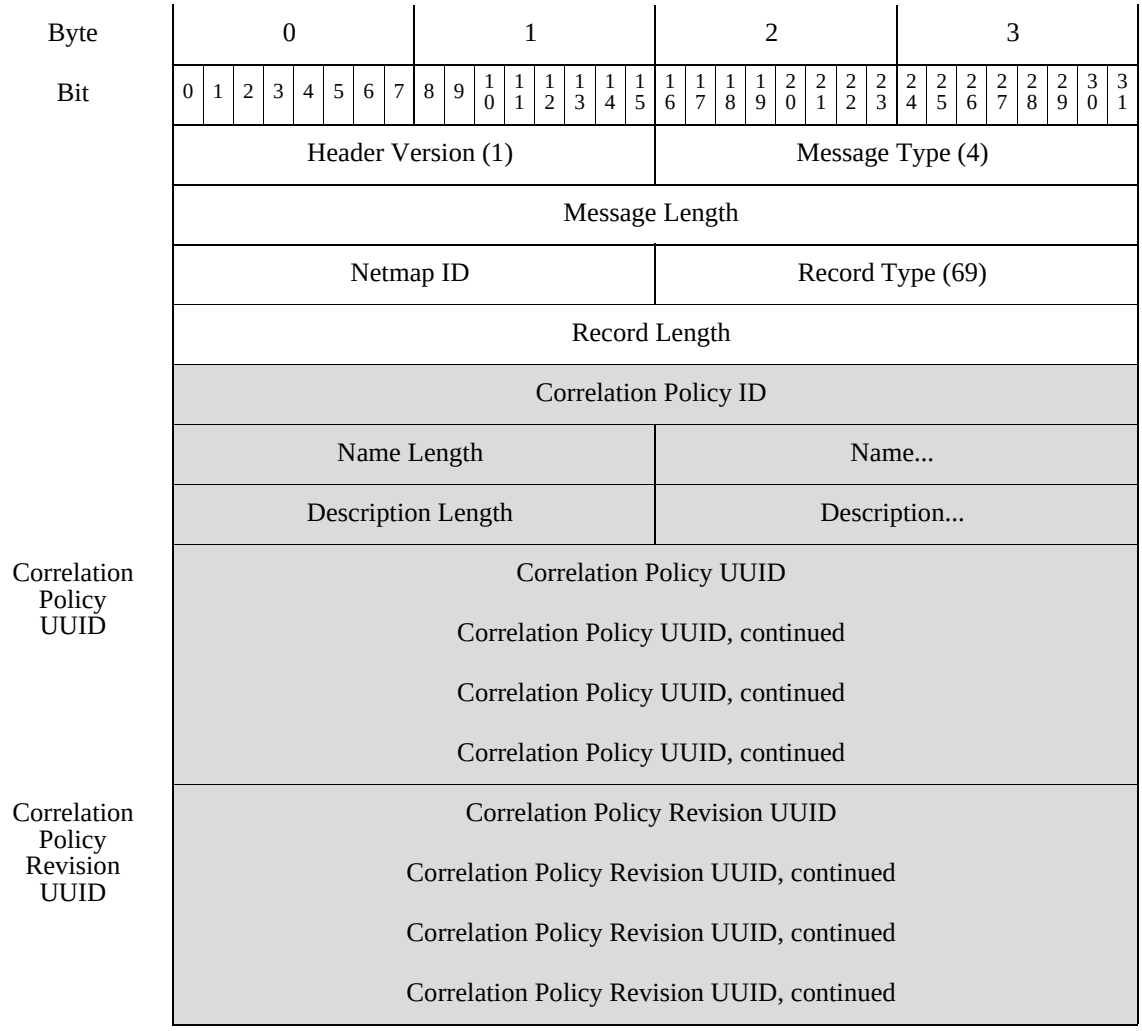
Field	Data Type	Description
Classification ID	uint32	The classification ID number.
Name Length	uint16	The number of bytes included in the name.
Name	string	The classification name.
Description Length	uint16	The number of bytes included in the description.
Description	string	The classification description.

Table 3-8 Classification Record Fields (continued)

Field	Data Type	Description
UUID	uint8[16]	A classification ID number that acts as a unique identifier for the classification.
Revision UUID	uint8[16]	A classification revision ID number that acts as a unique identifier for the classification revision.

Correlation Policy Record

The eStreamer service transmits metadata containing the correlation policy for a correlation event within a Correlation Policy record, the format of which is shown below. (Correlation policy information is sent when the Version 3 or Version 4 metadata flag—bit 15 or bit 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of 69, indicating a Correlation Policy record.



The following table describes the fields in the Correlation Policy record.

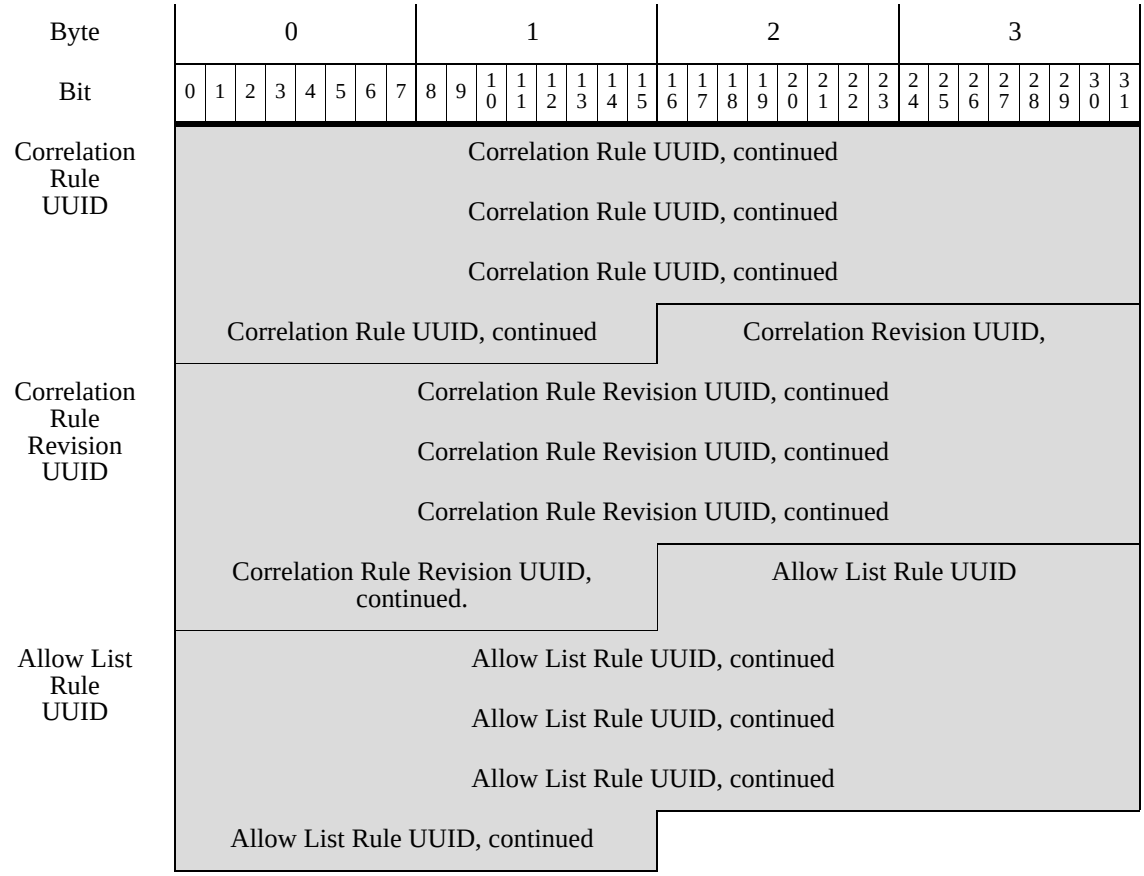
Table 3-9 Correlation Policy Record Fields

Field	Data Type	Description
Correlation Policy ID	uint32	The correlation policy ID number. This field is the unique key for this record.
Name Length	uint16	The number of bytes included in the correlation policy name.
Name	string	The name of the correlation policy that triggered the event.
Description Length	uint16	The number of bytes included in the correlation policy description.
Description	string	The description of the correlation policy that triggered the event.
UUID	uint8[16]	A correlation policy ID number that acts as a unique identifier for the correlation policy.
Revision UUID	uint8[16]	A correlation policy revision ID number that acts as a unique identifier for the correlation policy.

Correlation Rule Record

The eStreamer service transmits metadata containing information on the correlation rule that triggered a correlation event within a Correlation Rule record, the format of which is shown below. (Correlation rule information is sent when the Version 3 or Version 4 metadata flag—bit 15 or bit 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of 70, indicating a Correlation Rule record.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Header Version (1)																	Message Type (4)															
Message Length																																
Netmap ID																	Record Type (70)															
Record Length																																
Correlation Rule ID																																
Name Length																	Name...															
Name...																	Description Length															
Description...																																
Event Type Length																	Event Type...															
Event Type...																	Correlation Rule UUID															



The following table describes the fields in the Correlation Rule record.

Table 3-10 Correlation Rule Record Fields

Field	Data Type	Description
Correlation Rule ID	uint32	The correlation rule ID number. This field is the unique key for this record.
Name Length	uint16	The number of bytes included in the correlation rule name.
Name	string	The name of the correlation rule that triggered the event.
Description Length	uint16	The number of bytes included in the correlation rule description.
Description	string	The description of the correlation rule that triggered the event.
Event Type Length	uint16	The number of bytes included in the event type description.
Event Type	string	The description of the event that triggered the correlation rule.
UUID	uint8[16]	A correlation rule ID number that acts as a unique identifier for the correlation rule.
Revision UUID	uint8[16]	A correlation rule revision ID number that acts as a unique identifier for the correlation rule revision.
Allow List UUID	uint8[16]	A correlation ID number that acts as a unique identifier for the event sent as a result of an allow list violation.

Security Zone Name Record

The eStreamer service transmits metadata containing information on the name of the security zone associated with an intrusion event or connection event within a Security Zone Name record, the format of which is shown below. (Security zone information is sent when the Version 4 metadata flag—bit 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `115`, indicating a Security Zone Name record. It contains a UUID String data block, block type 14 in the series 2 set of data blocks.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Header Version (1)																	Message Type (4)															
Message Length																																
Netmap ID																	Record Type (115)															
Record Length																																
Security Zone Name Data Block (14)																																
Security Zone Name Data Block Length																																
Security Zone UUID																																
String Block Type (0)																																
String Block Length																																
Security Zone Name...																																

The following table describes the fields in the Security Zone Name data block.

Table 3-11 Security Zone Name Data Block Fields

Field	Data Type	Description
Security Zone Name Data Block Type	uint32	Initiates a Security Zone Name data block. This value is always `14`. The block type is a series 2 block.
Security Zone Name Data Block Length	uint32	Length of the data block. Includes the number of bytes of data plus the 8 bytes in the two data block header fields.
Security Zone UUID	uint8[16]	The unique identifier for the security zone associated with the connection event. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the security zone. This value is always `0`.

Table 3-11 Security Zone Name Data Block Fields (continued)

Field	Data Type	Description
String Block Length	uint32	The number of bytes included in the security zone name String data block, including eight bytes for the block type and header fields plus the number of bytes in the name.
Security Zone Name	string	The security zone name.

Interface Name Record

The eStreamer service transmits metadata containing information on the name of the interface associated with an intrusion event or connection event within an Interface Name record, the format of which is shown below. (Interface name information is sent when the Version 4 metadata flag—bit 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `116`, indicating an Interface Name record. It contains a UUID String data block, block type 14 in the series 2 set of data blocks.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Header Version (1)																	Message Type (4)															
Message Length																																
Netmap ID																	Record Type (116)															
Record Length																																
Interface Name Data Block (14)																																
Interface Name Data Block Length																																
Interface UUID																																
String Block Type (0)																																
String Block Length																																
Interface Name...																																

The following table describes the fields in the Interface Name data block.

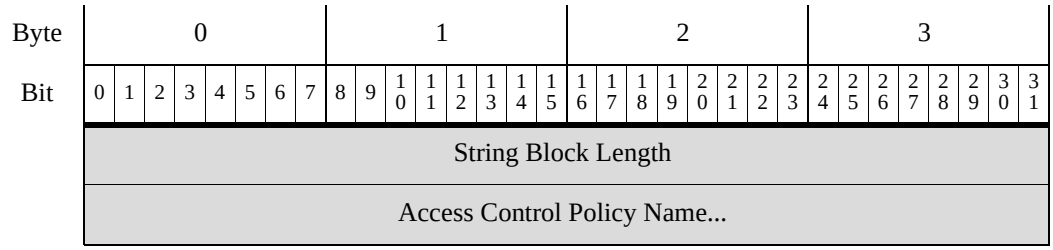
Table 3-12 Interface Name Data Block Fields

Field	Data Type	Description
Interface Name Data Block Type	uint32	Initiates an Interface Name data block. This value is always `14`. The block type is a series 2 block.
Interface Name Data Block Length	uint32	Length of the data block. Includes the number of bytes of data plus the 8 bytes in the two data block header fields.
Interface UUID	uint8[16]	An interface ID number that acts as a unique identifier for the interface associated with the connection event. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the interface. This value is always `0`.
String Block Length	uint32	The number of bytes included in the interface name String data block, including eight bytes for the block type and header fields plus the number of bytes in the interface name.
Interface Name	string	The interface name.

Access Control Policy Name Record

The eStreamer service transmits metadata on the name of the access control policy that triggered an intrusion event or connection event within an Access Control Policy Name record, the format of which is shown below. (Access control policy name information is sent when the Version 4 metadata flag—bit 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `117`, indicating an Access Control Policy Name record. It contains a UUID String data block, block type 14 in the series 2 set of data blocks.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Header Version (1)																	Message Type (4)															
Message Length																																
Netmap ID																	Record Type (117)															
Record Length																																
Access Control Policy Name Data Block (14)																																
Access Control Policy Name Data Block Length																																
Access Control Policy UUID																																
String Block Type (0)																																



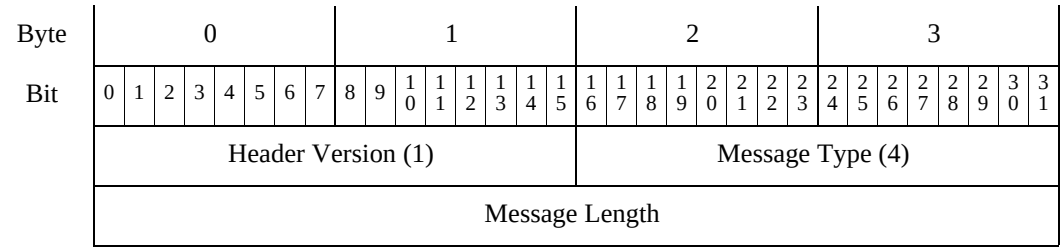
The following table describes the fields in the Access Control Policy Name data block.

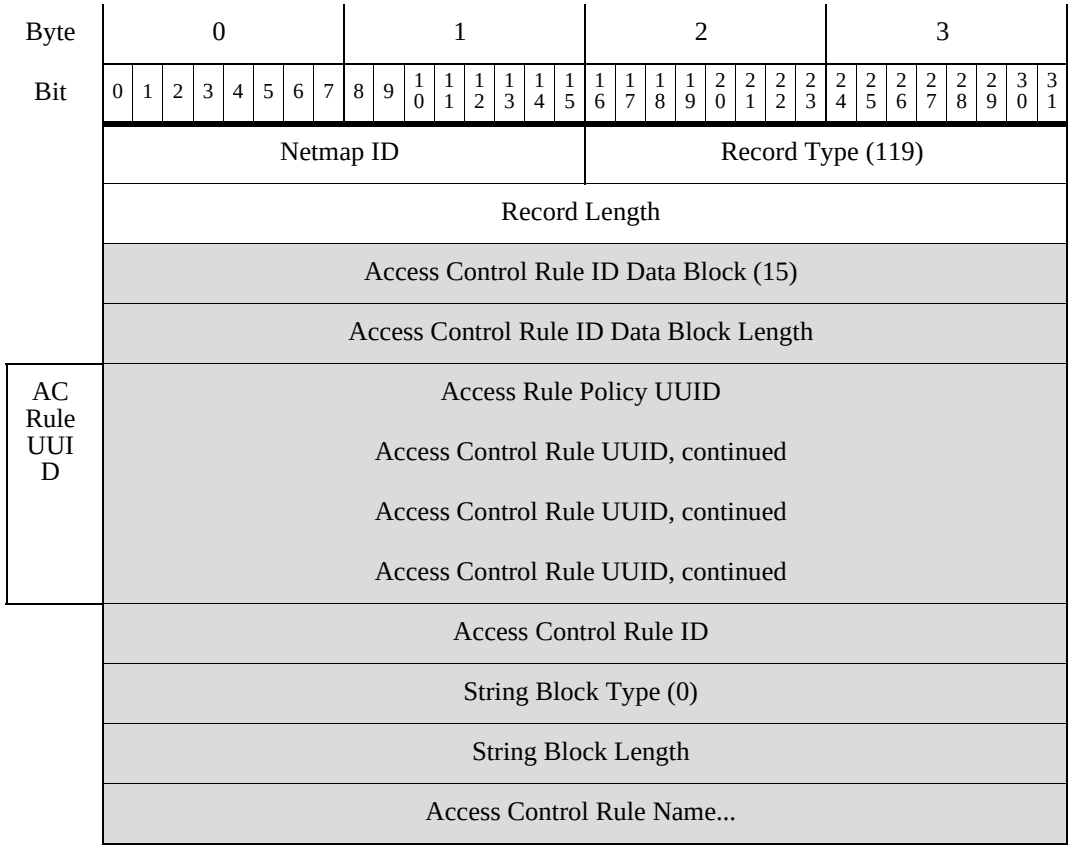
Table 3-13 Access Control Policy Name Data Block Fields

Field	Data Type	Description
Access Control Policy Name Data Block Type	uint32	Initiates an Access Control Policy Name data block. This value is always `14`. The block type is a series 2 block.
Access Control Policy Name Data Block Length	uint32	Length of the data block. Includes the number of bytes of data plus the 8 bytes in the two data block header fields.
Access Control Policy UUID	uint8[16]	An ID number that acts as a unique identifier for the access control policy associated with the intrusion event or connection event. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the access control policy. This value is always `0`.
String Block Length	uint32	The number of bytes included in the access control policy name String data block, including eight bytes for the block type and header fields plus the number of bytes in the access control policy name.
Access Control Policy Name	string	The access control policy name.

Access Control Rule ID Record Metadata

The eStreamer service transmits metadata containing information about the access control rule that triggered an intrusion event or connection event within an Access Control Rule ID record, the format of which is shown below. Access control rule metadata is sent when the Version 4 metadata flag—bit 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `119`, indicating an Access Control Rule ID record. It contains a Rule ID data block, block type 15 in the series 2 set of data blocks.





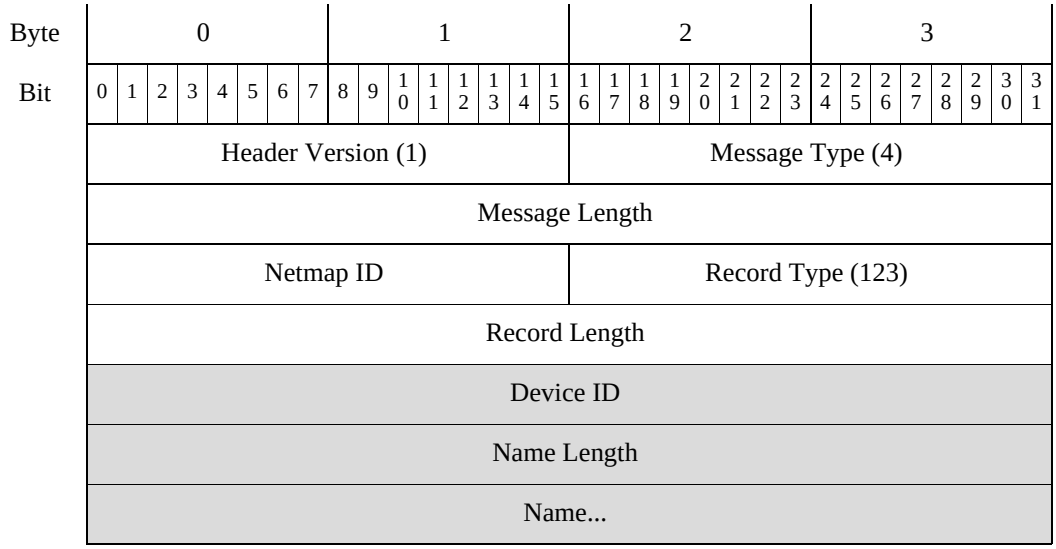
The following table describes the fields in the Access Control Rule ID data block.

Table 3-14 Access Control Rule ID Data Block Fields

Field	Data Type	Description
Access Control Rule ID Data Block Type	uint32	Initiates an Access Control Rule ID data block. This value is always `15`. The block type is a series 2 block.
Access Control Rule ID Data Block Length	uint32	Length of the data block. Includes the number of bytes of data plus the 8 bytes in the two data block header fields.
Access Control Rule UUID	uint8[16]	UUID of the Access Control Rule. This field, along with Access Control Rule ID, together are the unique key for this record.
Access Control Rule ID	uint32	The internal identifier for the rule in the access control policy associated with the connection event. This field, along with Access Control Rule UUID, together are the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the access control rule. This value is always `0`.
String Block Length	uint32	The number of bytes included in the String data block, including eight bytes for the block type and header fields plus the number of bytes in the rule name.
Access Control Rule Name	string	The access control rule name.

Managed Device Record Metadata

The eStreamer service transmits metadata containing information on the managed device associated with an intrusion event within a Managed Device record, the format of which is shown below. Managed device metadata is sent when the Version 4 metadata flag—bit 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `123`, indicating a Managed Device record.



The following table describes the fields in the Managed Device record.

Table 3-15 Managed Device Record Fields

Field	Data Type	Description
Device ID	uint32	ID number of the managed device. This field is the unique key for this record.
Name Length	uint32	The number of bytes included in the name.
Name	string	The managed device name.

Malware Event Record 5.1.1+

The fields in the malware event record are shaded in the following graphic. The record type is 125.

You request malware event records by setting the malware event flag—bit 30 in the Request Flags field—in the request message with an event version of `2` and an event code of `101`. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record. It contains a Malware Event data block, one of block types 24, 33, 35, 44, 47, or in the series 2 set of data blocks.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Header Version (1)																	Message Type (4)															
Message Length																																
Netmap ID																	Record Type (125)															
Record Length																																
eStreamer Server Timestamp (in events, only if bit 23 is set)																																
Reserved for Future Use (in events, only if bit 23 is set)																																
Malware Event Data Block																																

The following table describes each malware event record data field.

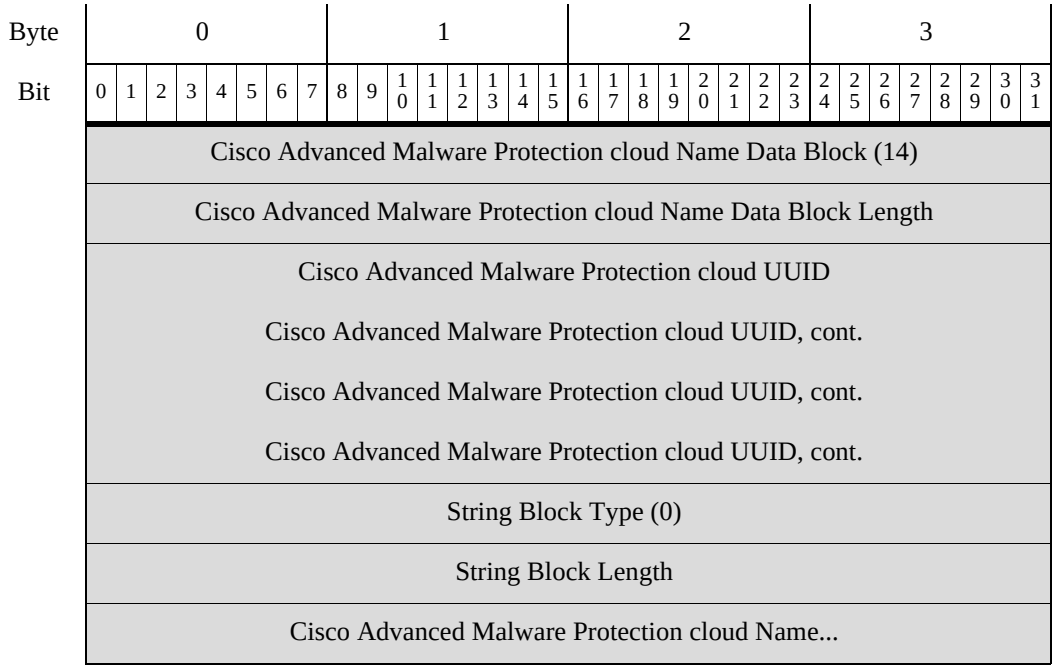
Table 3-16 Malware Event Record Fields

Field	Data Type	Description
Malware Event Data Block	variable	Indicates a malware event data block. See Malware Event Data Block 7.0+, page 3-89 for more information.

Cisco Advanced Malware Protection Cloud Name Metadata

The eStreamer service transmits metadata containing information on the name of the Cisco Advanced Malware Protection cloud (referred to as the AMP cloud or simply cloud) associated with an intrusion event or connection event within a Cisco Advanced Malware Protection cloud Name record, the format of which is shown below. (AMP cloud name information is sent when the Version 4 metadata flag—bit 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `127`, indicating a Cisco Advanced Malware Protection cloud Name record. It contains a UUID String data block, block type 14 in the series 2 set of data blocks.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Header Version (1)																	Message Type (4)															
Message Length																																
Netmap ID																	Record Type (127)															
Record Length																																



The following table describes the fields in the Cisco Advanced Malware Protection cloud Name data block.

Table 3-17 Cisco Advanced Malware Protection cloud Name Data Block Fields

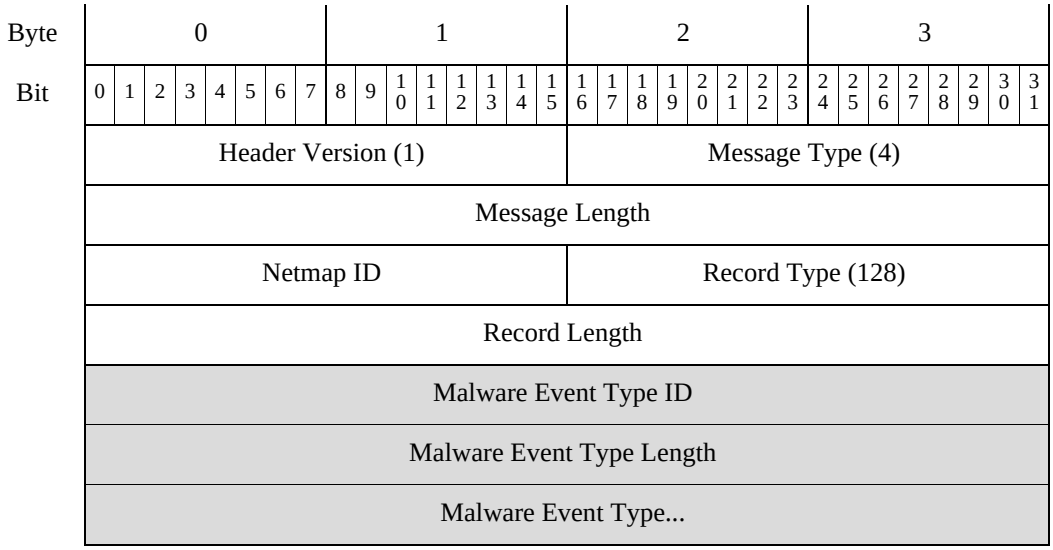
Field	Data Type	Description
Cisco Advanced Malware Protection cloud Name Data Block Type	uint32	Initiates a Cisco Advanced Malware Protection cloud Name data block. This value is always `14`. The block type is a series 2 block.
Cisco Advanced Malware Protection cloud Name Data Block Length	uint32	Length of the data block. Includes the number of bytes of data plus the 8 bytes in the two data block header fields.
Cisco Advanced Malware Protection cloud UUID	uint8[16]	A Cisco Advanced Malware Protection cloud ID number that acts as a unique identifier for the Cisco Advanced Malware Protection cloud associated with the connection event. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the Cisco Advanced Malware Protection cloud. This value is always `0`.

Table 3-17 Cisco Advanced Malware Protection cloud Name Data Block Fields (continued)

Field	Data Type	Description
String Block Length	uint32	The number of bytes included in the Cisco Advanced Malware Protection cloud Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Cisco Advanced Malware Protection cloud name.
Cisco Advanced Malware Protection cloud Name	string	The Cisco Advanced Malware Protection cloud name.

Malware Event Type Metadata

The eStreamer service transmits metadata containing malware event type information for an event within a malware event type record, the format of which is shown below. (Malware event type information is sent when the metadata flag, bit 20 in the request flags field of a request message, is set. See Request Flags, page 2-13.) Note that the record type field, which appears after the message length field, has a value of `128`, indicating a malware event type record.



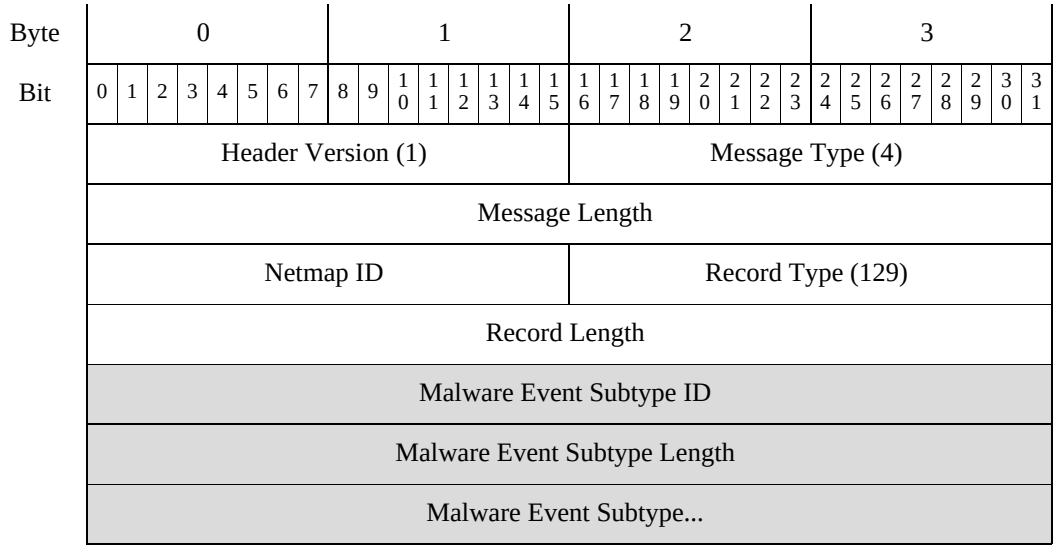
The following table describes the fields in the malware event type record.

Table 3-18 Malware Event Type Record Fields

Field	Data Type	Description
Malware Event Type ID	uint32	The malware event type ID number. This field is the unique key for this record.
Malware Event Type Length	uint32	The number of bytes included in the malware event type.
Malware Event Type	string	The type of malware event.

Malware Event Subtype Metadata

The eStreamer service transmits metadata containing malware event subtype information for an event within a malware event subtype record, the format of which is shown below. (Malware event type information is sent when the metadata flag, bit 20 in the request flags field of a request message, is set. See Request Flags, page 2-13.) Note that the record type field, which appears after the message length field, has a value of `129`, indicating a malware event subtype record.



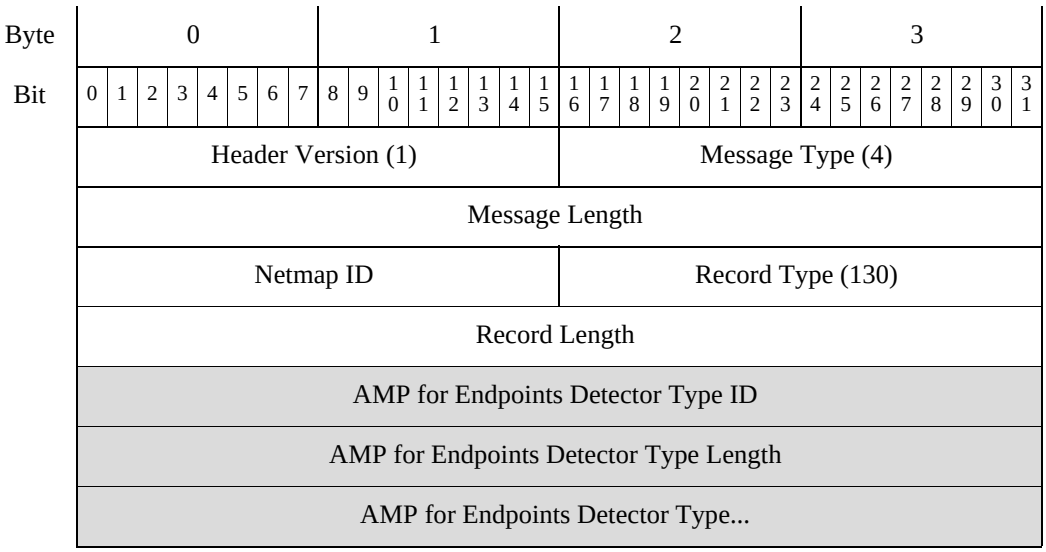
The following table describes the fields in the malware event subtype record.

Table 3-19 Malware Event Subtype Record Fields

Field	Data Type	Description
Malware Event Subtype ID	uint32	The malware event subtype ID number. This field is the unique key for this record.
Malware Event Subtype Length	uint32	The number of bytes included in the malware event subtype.
Malware Event Subtype	string	The malware event subtype.

AMP for Endpoints Detector Type Metadata

The eStreamer service transmits metadata containing AMP for Endpoints detector type information for an event within a AMP for Endpoints Detector Type record, the format of which is shown below. (AMP for Endpoints detector type information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `130`, indicating a AMP for Endpoints detector type record.



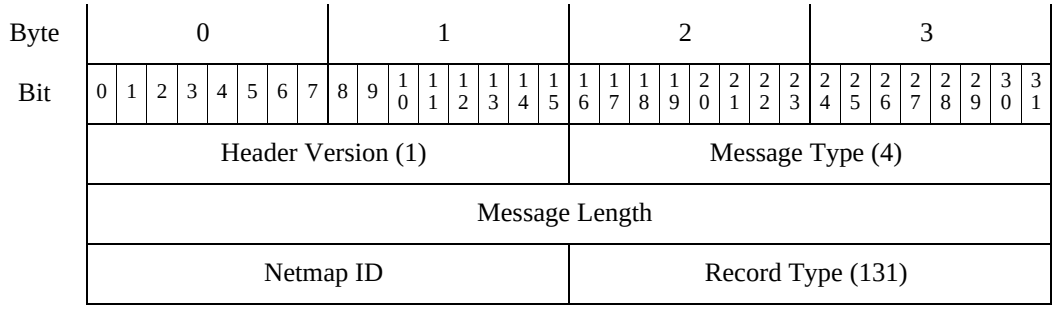
The following table describes the fields in the AMP for Endpoints Detector Type record.

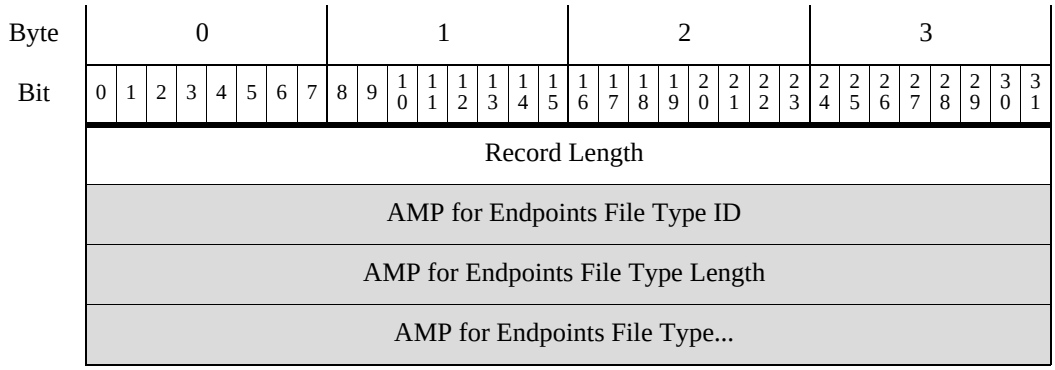
Table 3-20 AMP for Endpoints Detector Type Record Fields

Field	Data Type	Description
AMP for Endpoints Detector Type ID	uint32	The AMP for Endpoints detector type ID number. This field is the unique key for this record.
AMP for Endpoints Detector Type Length	uint32	The number of bytes included in the AMP for Endpoints detector type.
AMP for Endpoints Detector Type	string	The type of AMP for Endpoints detector.

AMP for Endpoints File Type Metadata

The eStreamer service transmits metadata containing AMP for Endpoints file type information for an event within a AMP for Endpoints File Type record, the format of which is shown below. (AMP for Endpoints file type information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `131`, indicating a AMP for Endpoints file type record.





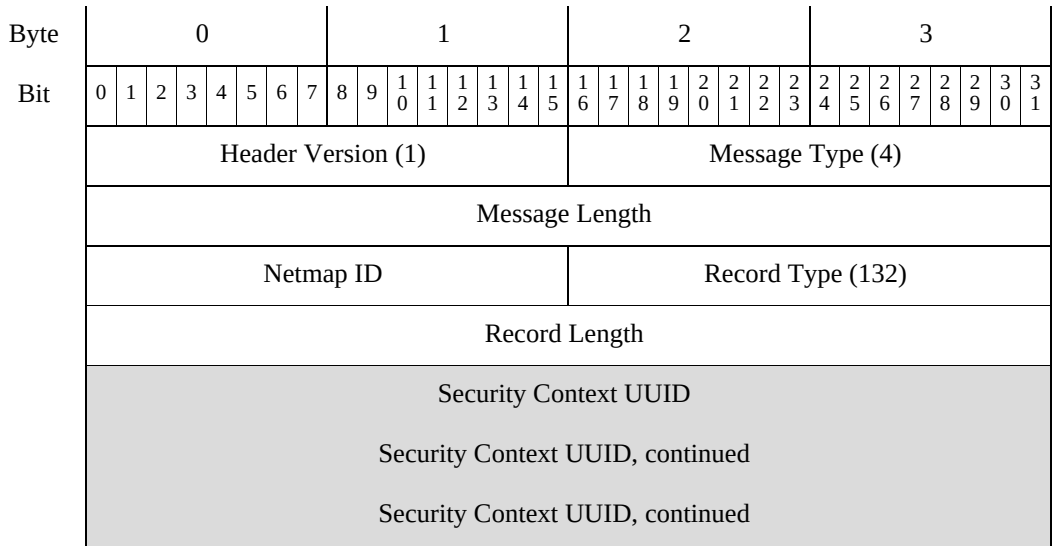
The following table describes the fields in the AMP for Endpoints File Type record.

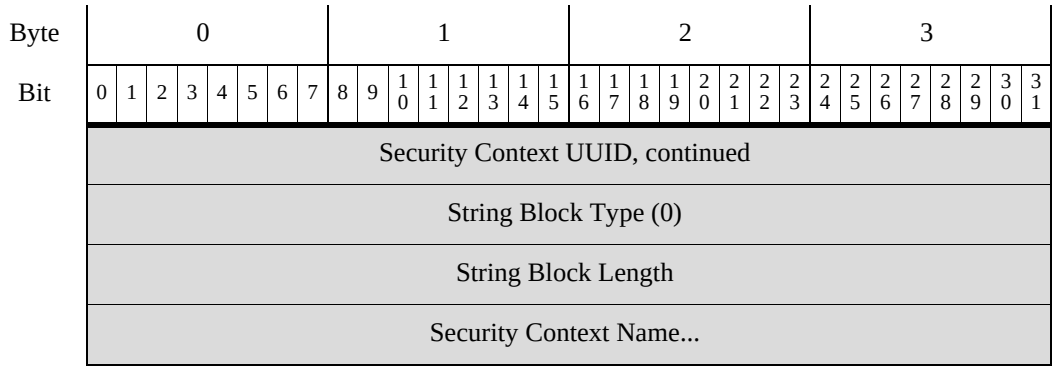
Table 3-21 AMP for Endpoints File Type Record Fields

Field	Data Type	Description
AMP for Endpoints File Type ID	uint32	The AMP for Endpoints file type ID number. This field is the unique key for this record.
AMP for Endpoints File Type Length	uint32	The number of bytes included in the AMP for Endpoints file type.
AMP for Endpoints File Type	string	The type of detected file.

Security Context Name

The eStreamer service transmits metadata containing Security Context Name information, the format of which is shown below. (Security Context Name information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `132`, indicating a Security Context Name record.





The following table describes the fields in the Security Context Name record.

Table 3-22 Security Context Name Record Fields

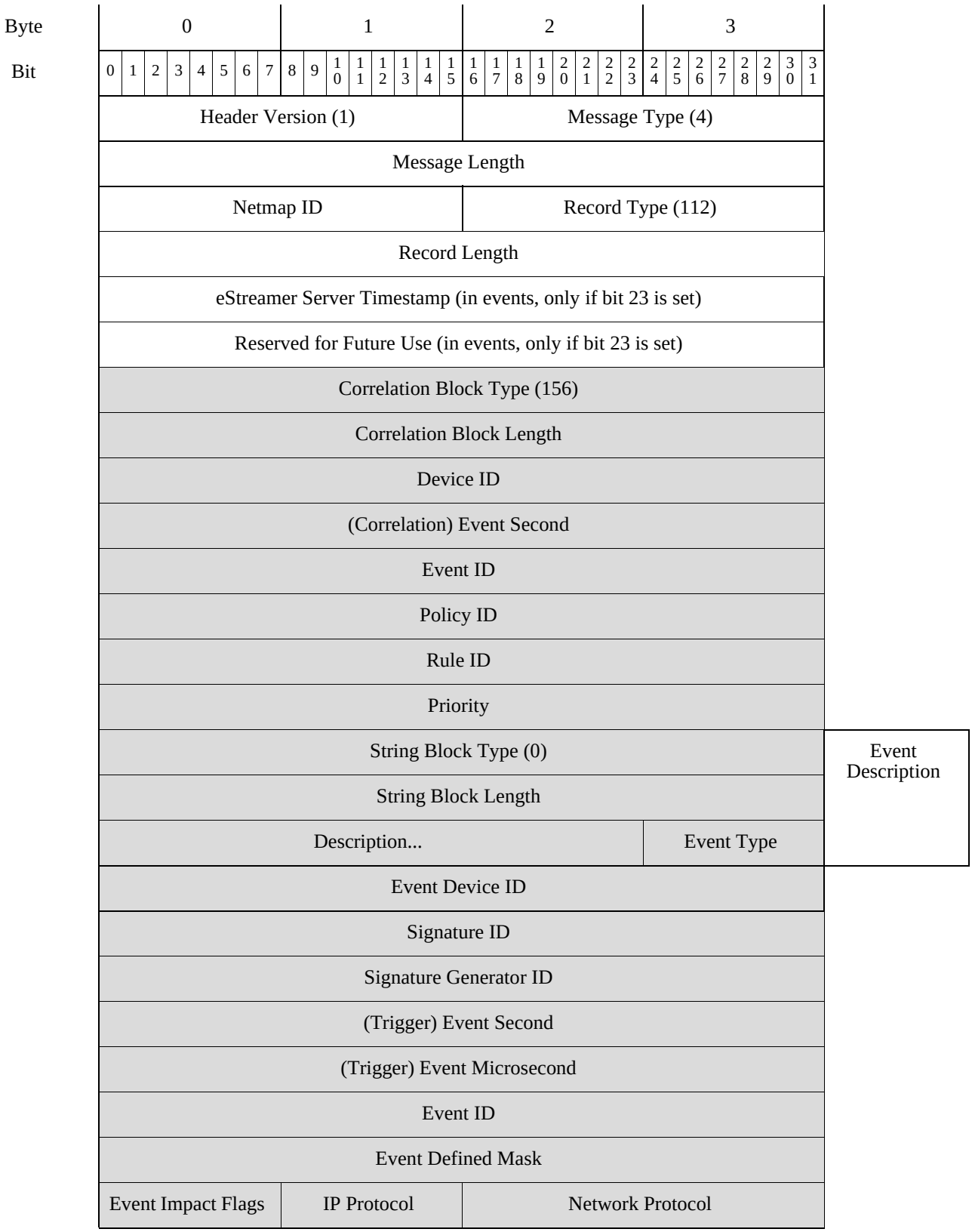
Field	Data Type	Description
Security Context UUID	uint8[16]	The UUID of the security context. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the security context. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Security Context Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Security Context name.
Security Context Name	string	The security context name.

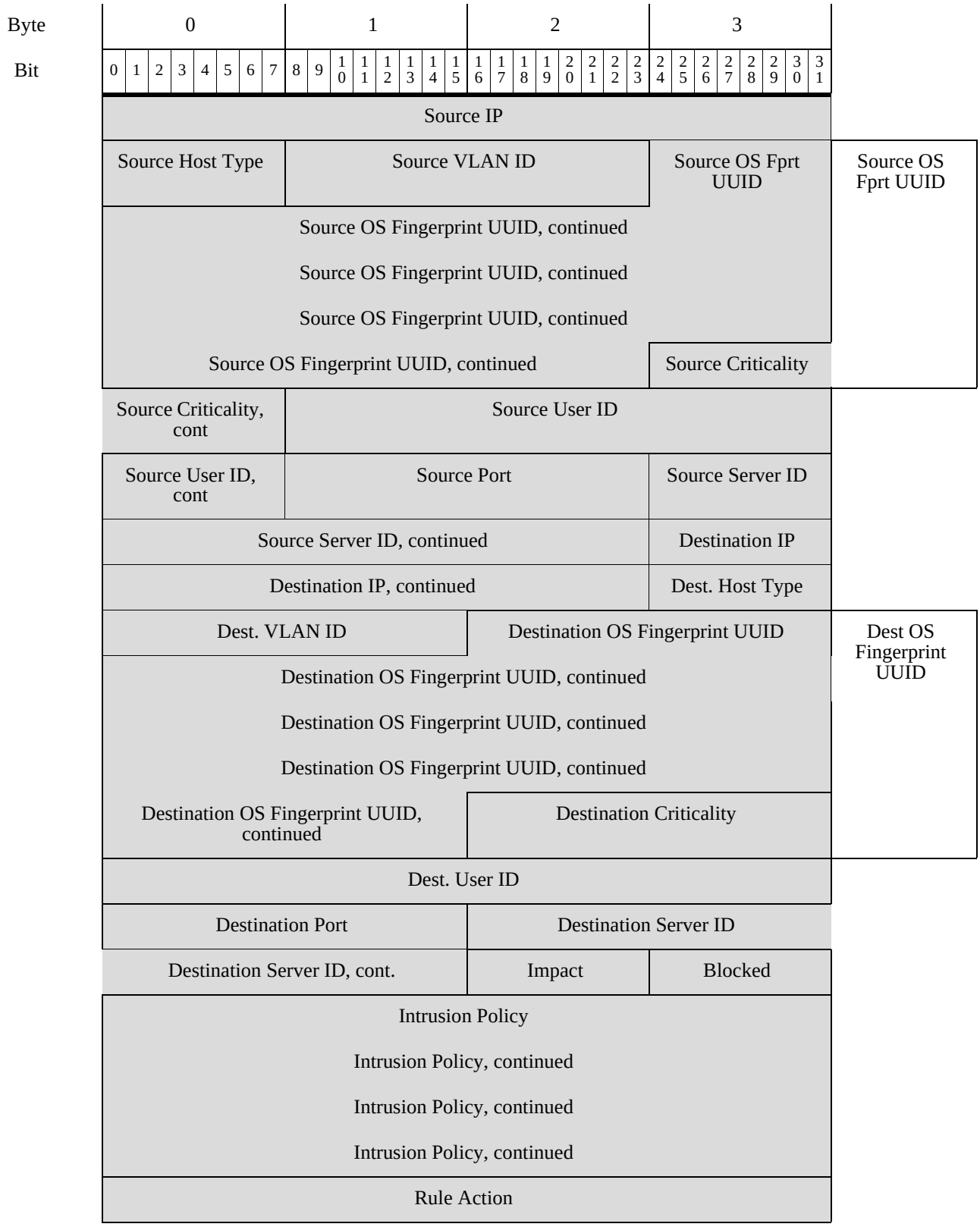
Correlation Event for 5.4+

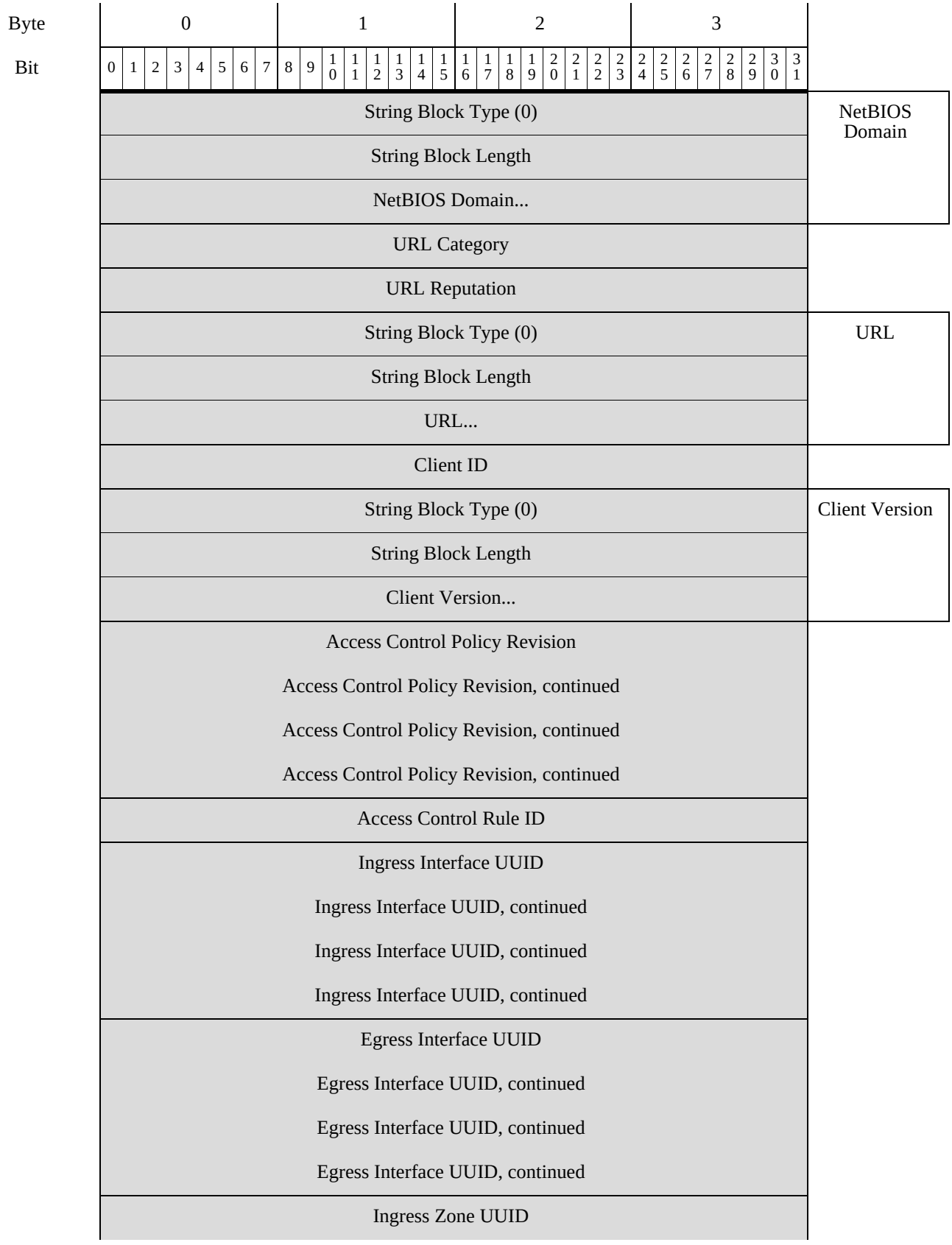
Correlation events (called compliance events in pre-5.0 versions) contain information about correlation policy violations. This message uses the standard eStreamer message header and specifies a record type of 112, followed by a correlation data block of type 156 in the series 1 set of data blocks. Data block type 156 differs from its predecessor (block type 128) in including IPv6 support.

The 5.4+ version of correlation events has new fields for geolocation, Security Intelligence, and SSL support.

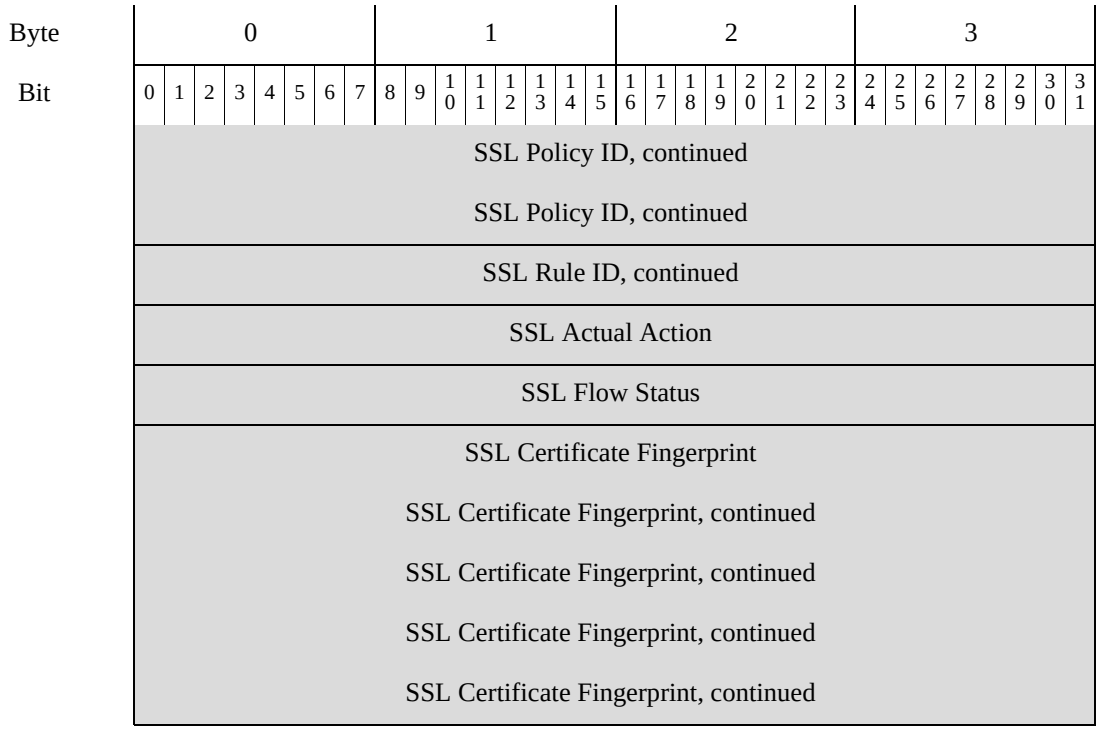
You can request 5.4+ correlation events from eStreamer only by extended request, for which you request event type code 31 and version code 9 in the Stream Request message (see Submitting Extended Requests, page 2-4 for information about submitting extended requests). You can optionally enable bit 23 in the flags field of the initial event stream request message, to include the extended event header. You can also enable bit 20 in the flags field to include user metadata.


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Header Version (1)
Message Type (4)
Message Length
Netmap ID
Record Type (112)
Record Length
eStreamer Server Timestamp (in events, only if bit 23 is set)
Reserved for Future Use (in events, only if bit 23 is set)
Correlation Block Type (156)
Correlation Block Length
Device ID
(Correlation) Event Second
Event ID
Policy ID
Rule ID
Priority
String Block Type (0)
String Block Length
Description...
Event Type
Event Description
Event Device ID
Signature ID
Signature Generator ID
(Trigger) Event Second
(Trigger) Event Microsecond
Event ID
Event Defined Mask
Event Impact Flags
IP Protocol
Network Protocol


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Source IP
Source Host Type
Source VLAN ID
Source OS Fprt UUID
Source OS Fprt UUID
Source OS Fingerprint UUID, continued
Source OS Fingerprint UUID, continued
Source OS Fingerprint UUID, continued
Source OS Fingerprint UUID, continued
Source Criticality
Source Criticality, cont
Source User ID
Source User ID, cont
Source Port
Source Server ID
Source Server ID, continued
Destination IP
Destination IP, continued
Dest. Host Type
Dest. VLAN ID
Destination OS Fingerprint UUID
Dest OS Fingerprint UUID
Destination OS Fingerprint UUID, continued
Destination OS Fingerprint UUID, continued
Destination OS Fingerprint UUID, continued
Destination OS Fingerprint UUID, continued
Destination Criticality
Dest. User ID
Destination Port
Destination Server ID
Destination Server ID, cont.
Impact
Blocked
Intrusion Policy
Intrusion Policy, continued
Intrusion Policy, continued
Intrusion Policy, continued
Rule Action


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
String Block Type (0)
String Block Length
NetBIOS Domain...
NetBIOS Domain
URL Category
URL Reputation
String Block Type (0)
String Block Length
URL...
URL
Client ID
String Block Type (0)
String Block Length
Client Version...
Client Version
Access Control Policy Revision
Access Control Policy Revision, continued
Access Control Policy Revision, continued
Access Control Policy Revision, continued
Access Control Rule ID
Ingress Interface UUID
Ingress Interface UUID, continued
Ingress Interface UUID, continued
Ingress Interface UUID, continued
Egress Interface UUID
Egress Interface UUID, continued
Egress Interface UUID, continued
Egress Interface UUID, continued
Ingress Zone UUID

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Ingress Zone UUID, continued																																
Ingress Zone UUID, continued																																
Ingress Zone UUID, continued																																
Egress Zone UUID																																
Egress Zone UUID, continued																																
Egress Zone UUID, continued																																
Egress Zone UUID, continued																																
Source IPv6 Address																																
Source IPv6 Address, continued																																
Source IPv6 Address continued																																
Source IPv6 Address, continued																																
Destination IPv6 Address																																
Destination IPv6 Address, continued																																
Destination IPv6 Address, continued																																
Destination IPv6 Address, continued																																
Source Country																	Destination Country															
Security Intelligence UUID																																
Security Intelligence UUID, continued																																
Security Intelligence UUID, continued																																
Security Intelligence UUID, continued																																
Security Context																																
Security Context, continued																																
Security Context, continued																																
Security Context, continued																																
SSL Policy ID																																
SSL Policy ID, continued																																



Note that the record structure includes a String block type, which is a block in series 1. For information about series 1 blocks, see Understanding Discovery (Series 1) Blocks, page 4-62.

Table 3-23 Correlation Event 5.4+ Data Fields

Field	Data Type	Description
Correlation Block Type	uint32	Indicates a correlation event data block follows. This field always has a value of 156. See Understanding Discovery (Series 1) Blocks, page 4-62.
Correlation Block Length	uint32	Length of the correlation data block, which includes 8 bytes for the correlation block type and length plus the correlation data that follows.
Device ID	uint32	Internal identification number of the managed device or Management Center that generated the correlation event. A value of zero indicates the Management Center. You can obtain managed device names by requesting Version 3 metadata. See Managed Device Record Metadata, page 3-33 for more information.
(Correlation) Event Second	uint32	UNIX timestamp indicating the time that the correlation event was generated (in seconds from 01/01/1970).
Event ID	uint32	Correlation event identification number.
Policy ID	uint32	Identification number of the correlation policy that was violated. See Service Record, page 4-15 for information about how to obtain policy identification numbers from the database.
Rule ID	uint32	Identification number of the correlation rule that triggered to violate the policy. See Service Record, page 4-15 for information about how to obtain policy identification numbers from the database.

Table 3-23 Correlation Event 5.4+ Data Fields (continued)

Field	Data Type	Description
Priority	uint32	Priority assigned to the event. This is an integer value from 0 to 5.
String Block Type	uint32	Initiates a string data block that contains the correlation violation event description. This value is always set to 0. For more information about string blocks, see String Data Block, page 4-71.
String Block Length	uint32	Number of bytes in the event description string block, which includes four bytes for the string block type and four bytes for the string block length, plus the number of bytes in the description.
Description	string	Description of the correlation event.
Event Type	uint8	Indicates whether the correlation event was triggered by an intrusion, host discovery, or user event: • 1 - intrusion • 2 - host discovery • 3 - user
Event Device ID	uint32	Identification number of the device that generated the event that triggered the correlation event. You can obtain device name by requesting Version 3 metadata. See Managed Device Record Metadata, page 3-33 for more information.
Signature ID	uint32	If the event was an intrusion event, indicates the rule identification number that corresponds with the event. Otherwise, the value is 0.
Signature Generator ID	uint32	If the event was an intrusion event, indicates the ID number of the Secure Firewall System preprocessor or rules engine that generated the event.
(Trigger) Event Second	uint32	UNIX timestamp indicating the time of the event that triggered the correlation policy rule (in seconds from 01/01/1970).
(Trigger) Event Microsecond	uint32	Microsecond (one millionth of a second) increment that the event was detected.
Event ID	uint32	Identification number of the event generated by the Cisco device.
Event Defined Mask	bits[32]	Set bits in this field indicate which of the fields that follow in the message are valid. See Table 3-21 on page 3-39 for a list of each bit value.

Table 3-23 Correlation Event 5.4+ Data Fields (continued)

Field	Data Type	Description
Event Impact Flags	bits[8]	Impact flag value of the event. The low-order eight bits indicate the impact level. Values are: • 0x01 (bit 0) — Source or destination host is in a network monitored by the system. • 0x02 (bit 1) — Source or destination host exists in the network map. • 0x04 (bit 2) — Source or destination host is running a server on the port in the event (if TCP or UDP) or uses the IP protocol. • 0x08 (bit 3) — There is a vulnerability mapped to the operating system of the source or destination host in the event. • 0x10 (bit 4) — There is a vulnerability mapped to the server detected in the event. • 0x20 (bit 5) — The event caused the managed device to drop the session (used only when the device is running in inline, switched, or routed deployment). Corresponds to blocked status in the Secure Firewall System web interface. • 0x40 (bit 6) — The rule that generated this event contains rule metadata setting the impact flag to red. The source or destination host is potentially compromised by a virus, trojan, or other piece of malicious software. • 0x80 (bit 7) — There is a vulnerability mapped to the client detected in the event. (version 5.0+ only) The following impact level values map to specific priorities on the Management Center. An `x` indicates the value can be 0 or 1: • gray (0, unknown): `00X00000` • red (1, vulnerable): `XXXX1XXX`, `XXX1XXXX`, `X1XXXXXX`, `1XXXXXXX` (version 5.0+ only) • orange (2, potentially vulnerable): `00X0011X` • yellow (3, currently not vulnerable): `00X0001X` • blue (4, unknown target): `00X00001`
IP Protocol	uint8	Identifier of the IP protocol associated with the event, if applicable.
Network Protocol	uint16	Network protocol associated with the event, if applicable.
Source IP Address	uint8[4]	This field is reserved but no longer populated. The Source IPv4 address is stored in the Source IPv6 Address field. See IP Addresses, page 1-4 for more information.
Source Host Type	uint8	Source host's type: • 0 — Host • 1 — Router • 2 — Bridge

Table 3-23 Correlation Event 5.4+ Data Fields (continued)

Field	Data Type	Description
Source VLAN ID	uint16	Source host's VLAN identification number, if applicable.
Source OS Fingerprint UUID	uint8[16]	A fingerprint ID number that acts a unique identifier for the source host's operating system. See Service Record, page 4-15 for information about obtaining the values that map to the fingerprint IDs.
Source Criticality	uint16	User-defined criticality value for the source host: • 0 — None • 1 — Low • 2 — Medium • 3 — High
Source User ID	uint32	Identification number for the user logged into the source host, as identified by the system.
Source Port	uint16	Source port in the event.
Source Server ID	uint32	Identification number for the server running on the source host.
Destination IP Address	uint8[4]	This field is reserved but no longer populated. The Destination IPv4 address is stored in the Destination IPv6 Address field. See IP Addresses, page 1-4 for more information.
Destination Host Type	uint8	Destination host's type: • 0 — Host • 1 — Router • 2 — Bridge
Destination VLAN ID	uint16	Destination host's VLAN identification number, if applicable.
Destination OS Fingerprint UUID	uint8[16]	A fingerprint ID number that acts as a unique identifier for the destination host's operating system. See Service Record, page 4-15 for information about obtaining the values that map to the fingerprint IDs.
Destination Criticality	uint16	User-defined criticality value for the destination host: • 0 — None • 1 — Low • 2 — Medium • 3 — High
Destination User ID	uint32	Identification number for the user logged into the destination host, as identified by the system.
Destination Port	uint16	Destination port in the event.
Destination Service ID	uint32	Identification number for the server running on the source host.

Table 3-23 Correlation Event 5.4+ Data Fields (continued)

Field	Data Type	Description
Impact	uint8	Impact flag value of the event. Values are: • 1 — Red (vulnerable) • 2 — Orange (potentially vulnerable) • 3 — Yellow (currently not vulnerable) • 4 — Blue (unknown target) • 5 — Gray (unknown impact)
Blocked	uint8	Value indicating what happened to the packed that triggered the intrusion event. • 0 — Pass • 1 — Dropped • 2 — Would be dropped (but not permitted by configuration) • 3— Partially dropped • 4— Block • 5— Would block • 6— Partial block • 7— Drop • 8— Would drop • 9— Reject • 10— Would reject • 11— React • 12— Would react • 13— Rewrite • 14— Would rewrite
Intrusion Policy	uint8[16]	UUID of the Intrusion Policy associated with the event.
Rule Action	uint32	The action selected in the user interface for the rule that triggered the event(allow, block, and so forth).
String Block Type	uint32	Initiates a string data block that contains the NetBIOS Domain. This value is always set to 0. For more information about string blocks, see String Data Block, page 4-71.
String Block Length	uint32	Number of bytes in the event description string block, which includes four bytes for the string block type and four bytes for the string block length, plus the number of bytes in the NetBIOS Domain.
NetBIOS Domain	string	Name of the NetBIOS Domain.
URL Category	uint32	The number designating the URL Category. See URL Category Record Metadata, page 4-24 for more information.
URL Reputation	uint32	ID number of the URL reputation. See URL Reputation Record Metadata, page 4-25

Table 3-23 Correlation Event 5.4+ Data Fields (continued)

Field	Data Type	Description
String Block Type	uint32	Initiates a string data block that contains the URL. This value is always set to 0. For more information about string blocks, see String Data Block, page 4-71.
String Block Length	uint32	Number of bytes in the event description string block, which includes four bytes for the string block type and four bytes for the string block length, plus the number of bytes in the URL.
URL	string	URL which triggered the correlation event.
Client ID	uint32	ID number of the client which detected the event.
String Block Type	uint32	Initiates a string data block that contains the Client Version. This value is always set to 0. For more information about string blocks, see String Data Block, page 4-71.
String Block Length	uint32	Number of bytes in the event description string block, which includes four bytes for the string block type and four bytes for the string block length, plus the number of bytes in the Client Version.
Client Version	string	Version of the client which detected the event.
Access Control Policy Revision	uint8[16]	Revision number of the rule associated with the triggered correlation event.
Access Control Rule ID	uint32	Internal identifier for the rule that triggered the event.
Ingress Interface UUID	uint8[16]	An interface ID that acts as the unique identifier for the ingress interface associated with correlation event.
Egress Interface UUID	uint8[16]	An interface ID that acts as the unique identifier for the egress interface associated with correlation event.
Ingress Zone UUID	uint8[16]	A zone ID that acts as the unique identifier for the ingress security zone associated with correlation event.
Egress Zone UUID	uint8[16]	A zone ID that acts as the unique identifier for the egress security zone associated with correlation event.
Source IPv6 Address	uint8[16]	IP address of the source host in the event, in IPv6 address octets.
Destination IPv6 Address	uint8[16]	IP address of the destination host in the event, in IPv6 address octets.
Source Country	uint16	Code for the country of the source host.
Destination Country	uint16	Code for the country of the destination host.
Security Intelligence UUID	uint8[16]	The UUID of the access control policy configured for Security Intelligence.
Security Context	uint8[16]	ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode.
SSL Policy ID	uint8[16]	ID number of the SSL policy that handled the connection.

Table 3-23 Correlation Event 5.4+ Data Fields (continued)

Field	Data Type	Description
SSL Rule ID	uint32	ID number of the SSL rule or default action that handled the connection.
SSL Actual Action	uint32	The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include: • 0 — 'Unknown' • 1 — 'Do Not Decrypt' • 2 — 'Block' • 3 — 'Block With Reset' • 4 — 'Decrypt (Known Key)' • 5 — 'Decrypt (Replace Key)' • 6 — 'Decrypt (Resign)'

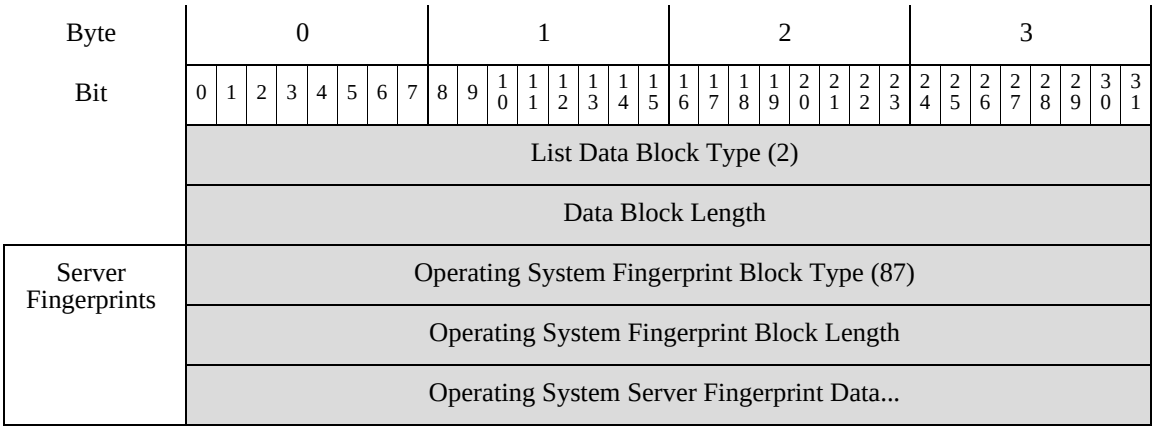
Table 3-23 Correlation Event 5.4+ Data Fields (continued)

Field	Data Type	Description
SSL Flow Status	uint32	Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include: • 0 — 'Unknown' • 1 — 'No Match' • 2 — 'Success' • 3 — 'Uncached Session' • 4 — 'Unknown Cipher Suite' • 5 — 'Unsupported Cipher Suite' • 6 — 'Unsupported SSL Version' • 7 — 'SSL Compression Used' • 8 — 'Session Undecryptable in Passive Mode' • 9 — 'Handshake Error' • 10 — 'Decryption Error' • 11 — 'Pending Server Name Category Lookup' • 12 — 'Pending Common Name Category Lookup' • 13 — 'Internal Error' • 14 — 'Network Parameters Unavailable' • 15 — 'Invalid Server Certificate Handle' • 16 — 'Server Certificate Fingerprint Unavailable' • 17 — 'Cannot Cache Subject DN' • 18 — 'Cannot Cache Issuer DN' • 19 — 'Unknown SSL Version' • 20 — 'External Certificate List Unavailable' • 21 — 'External Certificate Fingerprint Unavailable' • 22 — 'Internal Certificate List Invalid' • 23 — 'Internal Certificate List Unavailable' • 24 — 'Internal Certificate Unavailable' • 25 — 'Internal Certificate Fingerprint Unavailable' • 26 — 'Server Certificate Validation Unavailable' • 27 — 'Server Certificate Validation Failure' • 28 — 'Invalid Action'
SSL Certificate Fingerprint	uint8[20]	SHA1 hash of the SSL Server certificate.

Understanding Series 2 Data Blocks

Beginning in version 4.10.0, the eStreamer service uses a second series of data blocks to package certain records such as intrusion event extra data. See Table 3-24 on page 3-53 for a list of all block types in the series. Series 2 blocks, like series 1 blocks, support variable-length fields and hierarchies of nested blocks. The series 2 block types include primitive blocks that provide the same mechanism for encapsulating nested inner blocks as the series 1 primitive block types. However, series 2 blocks and series 1 blocks have separate numbering systems.

The following example shows the how primitive blocks are used. The list data block (series 2 block type 31) defines an array of operating system fingerprints (each of which is a type 87 block itself with variable length). The overall type 31 data block length is self-describing via the Data Block Length field, which contains the length of the data portion of the message, excluding the 8 bytes in the block type and block length fields.



In the following table, the Data Block Status field indicates whether the block is current (the latest version) or legacy (used in an older version and can still be requested through eStreamer).

Table 3-24 Series 2 Block Types

Type	Content	Data Block Status	Description
0	String	Current	Encapsulates variable string data. See String Data Block, page 3-57 for more information.
1	BLOB	Current	Encapsulates binary data and is used specifically for banners. See BLOB Data Block, page 3-58 for more information.
2	List	Current	Encapsulates a list of other data blocks. See List Data Block, page 3-59 for more information.
3	Generic List	Current	Encapsulates a list of other data blocks. For deserialization, it is the equivalent of the List data block. See Generic List Data Block, page 3-60 for more information.
4	Event Extra Data	Legacy	Contains intrusion event extra data. See Intrusion Event Extra Data Record, page B-65 for more information.

Table 3-24 Series 2 Block Types (continued)

Type	Content	Data Block Status	Description
5	Extra Data Type	Current	Contains extra data metadata. See Intrusion Event Extra Data Metadata, page B-66 for more information.
14	UUID String Mapping	Current	Block used by various metadata messages to map UUID values to descriptive strings. See UUID String Mapping Data Block, page 3-60.
15	Access Control Policy Rule ID Metadata	Current	Contains metadata for access control rules. See Access Control Policy Rule ID Metadata Block, page 3-63.
16	Malware Event	Legacy	Contains information on malware events, such as the malware detected or quarantined within a Cisco Advanced Malware Protection cloud, the detection method, and hosts and users affected by the malware. See Malware Event Data Block 5.1, page B-68. Deprecated by block 24, Malware Event Data Block 5.3.1, page B-92.
19	ICMP Type Data Block	Current	Contains metadata describing ICMP types. See ICMP Type Data Block, page 3-64.
20	ICMP Code Data Block	Current	Contains metadata describing ICMP codes. See ICMP Code Data Block, page 3-65.
21	Access Control Policy Rule Reason Data Block	Current	Contains information explaining access control policy rule reasons. See Access Control Policy Rule Reason Data Block for 6.0+, page 3-75.
22	IP Reputation Category Data Block	Current	Contains information on IP reputation categories explaining why an IP address was blocked. See Access Control Policy Name Data Block, page 3-77.
23	File Event	Legacy	Contains information on file events, such as the source, SHA hash, and the disposition of the file. See File Event for 5.1.1.x, page B-287. It is superseded by block 32, Access Control Policy Rule ID Metadata Block, page 3-63.
24	Malware Event	Legacy	Contains information on malware events, such as the malware detected or quarantined within a Cisco Advanced Malware Protection cloud, the detection method, and hosts and users affected by the malware. See Malware Event Data Block 5.1.1.x, page B-72. Deprecates block 16, Malware Event Data Block 5.1, page B-68. Deprecated by block 33, Malware Event Data Block 5.3.1, page B-92.
25	Intrusion Event	Legacy	Contains information on intrusion events, including information to match intrusion events with connection and malware events. See Intrusion Event Record 5.1.1.x, page B-23. Deprecated by block 34, Intrusion Event Record 5.2.x, page B-12.
26	File Event SHA Hash	Legacy	Contains the SHA hash and name of files that have been identified as containing malware. See File Event SHA Hash for 5.1.1-5.2.x, page B-327. Deprecated by block 40, File Event SHA Hash for 5.3+, page 3-100.

Table 3-24 Series 2 Block Types (continued)

Type	Content	Data Block Status	Description
27	Rule Documentation Data Block	Current	Contains information about rules used to generate events. See Rule Documentation Data Block for 5.2+, page 3-103 for more information.
28	Geolocation Data Block	Current	Contains country codes and associated country name. See Geolocation Data Block for 5.2+, page 3-110.
32	File Event	Legacy	Contains information on file events, such as the source, SHA hash, and the disposition of the file. See File Event for 5.2.x, page B-291. It deprecates File Event for 5.1.1.x, page B-287. Deprecated by block 38, File Event for 5.3, page B-295.
33	Malware Event	Current	Contains information on malware events, such as the malware detected or quarantined within a Cisco Advanced Malware Protection cloud, the detection method, and hosts and users affected by the malware. See Malware Event Data Block 5.2.x, page B-78. Deprecates block 24, Malware Event Data Block 5.1.1.x, page B-72. Deprecated by block 35, Malware Event Data Block 5.3, page B-85.
34	Intrusion Event	Legacy	Contains information on intrusion events, including information to match intrusion events with connection and malware events. See Intrusion Event Record 5.2.x, page B-12. Deprecates block 25. Deprecated by block 41, Intrusion Event Record 5.3, page B-17.
35	Malware Event	Legacy	Contains information on malware events, including IOC information. See Malware Event Data Block 5.3, page B-85. Deprecates block 33, Malware Event Data Block 5.2.x, page B-78. Deprecated by block 44, Malware Event Data Block 5.3, page B-85.
38	File Event	Legacy	Contains information on file events, such as the source, SHA hash, and the disposition of the file. See File Event for 5.3, page B-295. It deprecates block 32. Deprecated by block 43, Malware Event Data Block 7.0+, page 3-89.
39	IOC Name Data Block	Current	Contains information about IOCs. See IOC Name Data Block for 5.3+, page 4-35
40	File Event SHA Hash	Current	Contains the SHA hash and name of files that have been identified as containing malware. See File Event SHA Hash for 5.3+, page 3-100. Deprecates block 26, File Event SHA Hash for 5.1.1-5.2.x, page B-327.
41	Intrusion Event	Legacy	Contains information on intrusion events, including information to match intrusion events with IOCs. See Intrusion Event Record 5.3, page B-17. Deprecates block 34. Deprecated by block 42, Intrusion Event Record 5.3.1, page B-29.

Table 3-24 Series 2 Block Types (continued)

Type	Content	Data Block Status	Description
42	Intrusion Event	Legacy	Contains information on intrusion events, including information to match intrusion events with IOCs. See Intrusion Event Record 5.3.1, page B-29. Deprecates block 41, Intrusion Event Record 5.3, page B-17. Deprecated by block 45, Intrusion Event Record 5.4.x, page B-36.
43	File Event	Legacy	Contains information on file events, such as the source, SHA hash, and the disposition of the file. See File Event for 5.3.1, page B-301. Deprecates block 38, File Event for 5.3, page B-295. Deprecated by block 46, File Event for 7.0+, page 3-79
44	Malware Event	Legacy	Contains information on malware events, including IOC information. See Malware Event Data Block 7.0+, page 3-89. Deprecates block 35, Malware Event Data Block 5.3, page B-85. Deprecated by block 47, Malware Event Data Block 7.0+, page 3-89
45	Intrusion Event	Legacy	Contains information on intrusion events. See Intrusion Event Record 5.4.x, page B-36. Deprecates block 42, Intrusion Event Record 5.3.1, page B-29. Deprecated by block 60, Intrusion Event Record 6.x, page B-44.
46	File Event	Legacy	Contains information on file events, such as the source, SHA hash, and the disposition of the file. See Malware Event Data Block 7.0+, page 3-89. Deprecates block 43, File Event for 5.3.1, page B-301.
47	Malware Event	Current	Contains information on malware events, including IOC information. See Malware Event Data Block 7.0+, page 3-89. Deprecates block 44, Malware Event Data Block 5.3.1, page B-92.
50	SSL Certificate Details	Current	Contains information regarding an SSL certificate. SeeSSL Certificate Details Data Block for 5.4+, page 3-121
51	SSL Rule ID	Current	Contains information about SSL Rules. See SSL Rule ID, page 3-114
56	File Event	Legacy	Contains information on file events. See File Event for 6.x, page B-317. Deprecates block 46, File Event for 5.4.x, page B-307. It is deprecated by block type 79, Malware Event Data Block 7.0+, page 3-89
57	User Record	Current	Contains information about users. See User Record, page 3-21
58	Endpoint Profile	Current	Contains information about network endpoints. See Endpoint Profile Data Block for 6.0+, page 3-68
59	Access Control Policy Rule Reason	Current	Contains information about access control policy rules. See Access Control Policy Rule Reason Data Block for 6.0+, page 3-75

Table 3-24 Series 2 Block Types (continued)

Type	Content	Data Block Status	Description
60	Intrusion Event	Legacy	Contains information on intrusion events. See Intrusion Event Record 6.x, page B-44. Deprecates block 45, Intrusion Event Record 5.3.1, page B-29. Deprecated by block 81, Intrusion Event Record 7.1+, page 3-7.
61	Name Description Mapping	Current	Used to map names to descriptions in many situations. See Name Description Mapping Data Block, page 3-61
62	Malware Event	Legacy	Contains information on malware events. See Malware Event Data Block 6.x, page B-109. Deprecates block 44, Malware Event Data Block 5.3.1, page B-92. Deprecated by block type 80, Malware Event Data Block 7.0+, page 3-89
64	Access Control Policy Name	Current	Contains information about access control policy names. See Access Control Policy Name Data Block, page 3-77
79	File Event	Current	Contains information on file events. See File Event for 7.0+, page 3-79. Deprecates block 56, File Event for 6.x, page B-317.
80	Malware Event	Current	Contains information on malware events. See Malware Event Data Block 7.0+, page 3-89. Deprecates block 62, Malware Event Data Block 6.x, page B-109.
81	Intrusion Event	Current	Contains information on intrusion events. See Intrusion Event Record 7.1+, page 3-7. Deprecates block 60, Intrusion Event Record 6.x, page B-44.

Series 2 Primitive Data Blocks

Both series 2 and series 1 blocks include a set of primitives that are used to encapsulate lists of variable-length blocks as well as variable-length strings and BLOBs within messages. These primitive blocks have the standard eStreamer block header discussed above in Data Block Header, page 2-26, but they appear only within other data blocks. Any number can be included in a given block type. For details on the structure of these blocks, see the following:

- String Data Block, page 3-57
- BLOB Data Block, page 3-58
- List Data Block, page 3-59
- Generic List Data Block, page 3-60
- UUID String Mapping Data Block, page 3-60
- Name Description Mapping Data Block, page 3-61

String Data Block

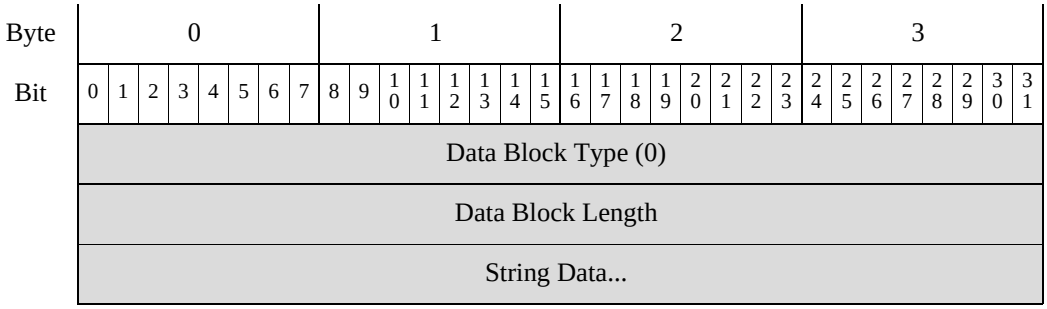
The eStreamer service uses the String data block to send string data in messages. These blocks commonly appear within other data blocks to identify, for example, operating system or server names.

Empty String data blocks (containing no data, only the header fields) have a block length of 8. eStreamer uses an empty String data block when it has no content for a string value, as might happen, for example, in the OS vendor string field in an Operating System data block when the vendor of the operating system is unknown.

The String data block has a block type of 0 in the series 2 group of blocks.

Note Strings returned in this data block are not always null-terminated (that is, the string characters are not always followed by a 0).

The following diagram shows the format of the String data block:



The following table describes the fields of the String data block.

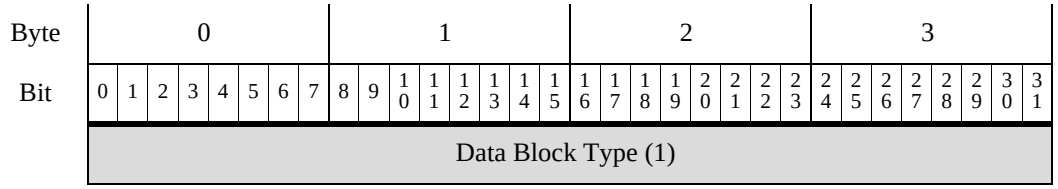
Table 3-25 String Block Fields

Field	Data Type	Description
Data Block Type	uint32	Initiates a String data block. This value is always `0`.
Data Block Length	uint32	Combined length in bytes of the string data block header and string data.
String Data	string	Contains the string data and may contain a terminating character (null byte) at the end of the string.

BLOB Data Block

The eStreamer service uses the BLOB data block to convey binary data. For example, host discovery records use the BLOB block to hold captured server banners. The BLOB data block has a block type of 1 in the series 2 group of blocks.

The following diagram shows the format of the BLOB data block:



Data Block Length
Binary Data...

The following table describes the fields of the BLOB data block.

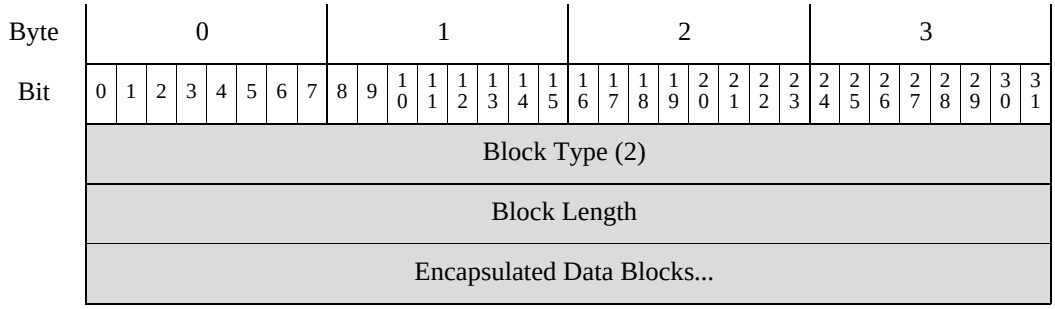
Table 3-26 BLOB Data Block Fields

Field	Data Type	Description
Data Block Type	uint32	Initiates a BLOB data block. This value is always `1`.
Data Block Length	uint32	Number of bytes in the BLOB data block, including eight bytes for the BLOB block type and length fields, plus the length of the binary data that follows.
Binary Data	variable	Contains binary data such as a server banner.

List Data Block

The eStreamer service uses the List data block to encapsulate a list of data blocks. For example, eStreamer can use the List data block to send a list of TCP servers, each of which is itself a data block. The List data block has a block type of 2 in the series 2 group of blocks.

The following diagram shows the basic format of a List data block:



The following table describes the fields of the List data block.

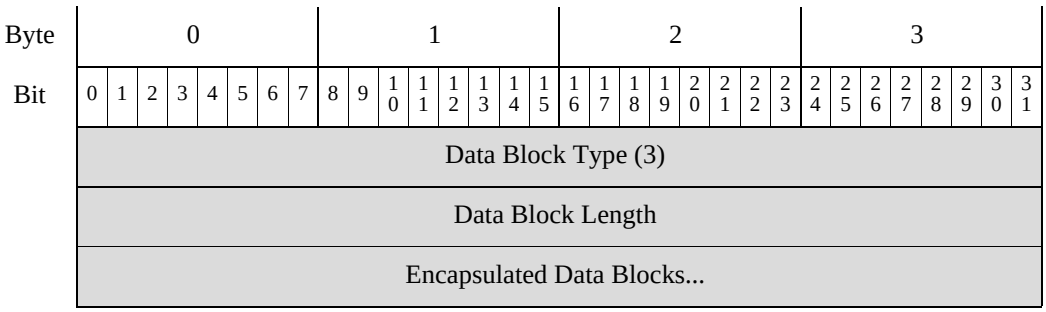
Table 3-27 List Data Fields

Field	Data Type	Description
Block Type	uint32	Initiates a List data block. This value is always `2`.
Block Length	uint32	Number of bytes in the List block and encapsulated data. For example, if there were three Sub-Server data blocks included in the list, the value here would include the total number of bytes in the Sub-Server blocks, plus eight bytes for the List block header.
Encapsulated Data Blocks	variable	Encapsulated data blocks up to the maximum number of bytes in the list block length.

Generic List Data Block

The eStreamer service uses the Generic List data block to encapsulate a list of data blocks. For example, the Host Profile data block contains information about multiple client applications and uses the Generic List block to embed a list of Client Application data blocks in the message. The Generic List data block has a block type of 3 in the series 2 group of blocks.

The following diagram shows the basic structure of a Generic List data block:



The following table describes the fields of the Generic List data block.

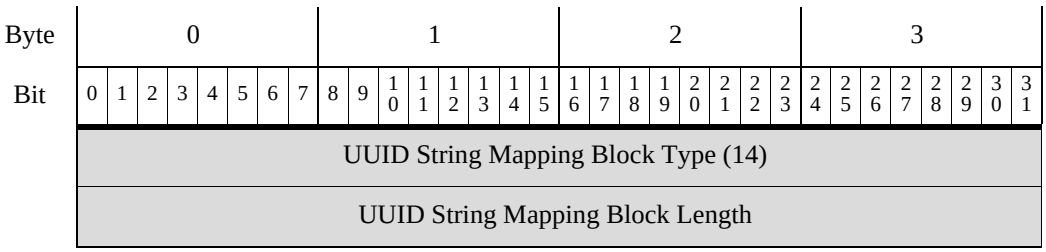
Table 3-28 Generic List Data Block Fields

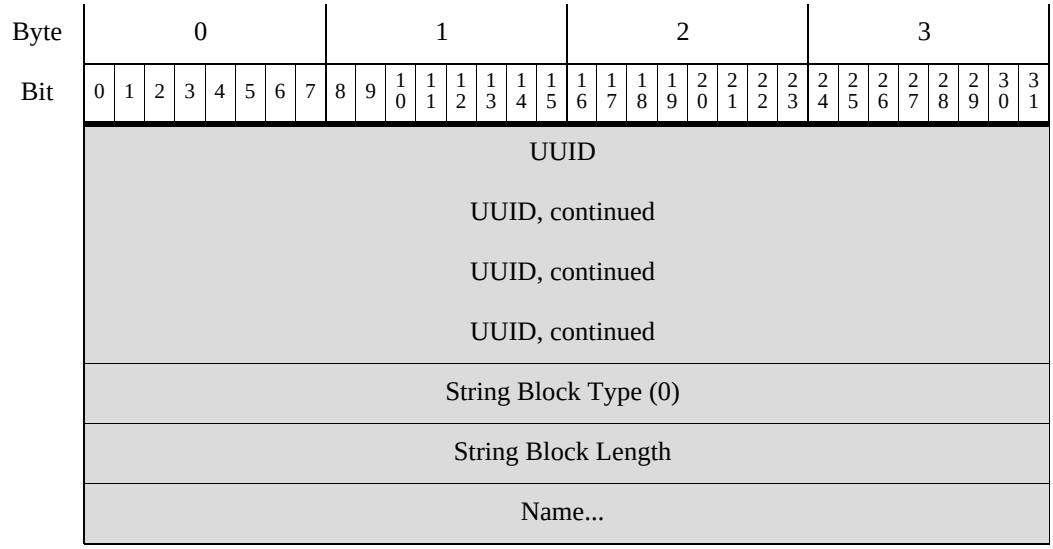
Field	Number of Bytes	Description
Data Block Type	uint32	Initiates a Generic List data block. This value is always 3.
Data Block Length	uint32	Number of bytes in the Generic List block and encapsulated data blocks. This number includes the eight bytes of the generic list block header fields, plus the total number of bytes in all of the encapsulated data blocks.
Encapsulated Data Blocks	variable	Encapsulated data blocks up to the maximum number of bytes in the Generic List block length.

UUID String Mapping Data Block

The eStreamer service uses the UUID String Mapping data block in various metadata messages to map UUID values to descriptive strings. The UUID String Mapping data block has a block type of 14 in series 2.

The following diagram shows the structure of the UUID String Mapping data block.





The following table describes the fields in the UUID String Mapping data block.

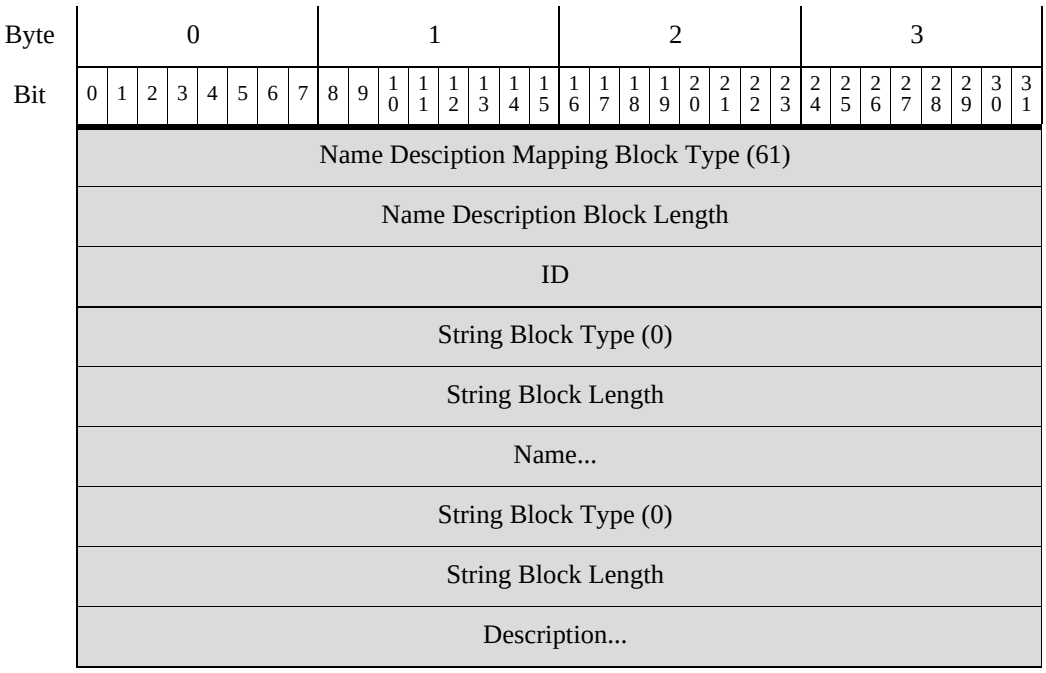
Table 3-29 UUID String Mapping Data Block Fields

Field	Data Type	Description
UUID String Mapping Block Type	uint32	Initiates a UUID String Mapping block. This value is always `14`.
UUID String Mapping Block Length	uint32	Total number of bytes in the UUID String Mapping block, including eight bytes for the UUID String Mapping block type and length fields, plus the number of bytes of data that follows.
UUID	uint8[16]	The unique identifier for the event or other object the UUID identifies. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the descriptive name associated with the UUID. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Name field.
Name	string	The descriptive name.

Name Description Mapping Data Block

The eStreamer service uses the Name Description Mapping data block in various metadata messages to map ID values to names and descriptive strings. The Name Description Mapping data block has a block type of 61 in series 2.

The following diagram shows the structure of the Name Description Mapping data block.



The following table describes the fields in the Name Description Mapping data block.

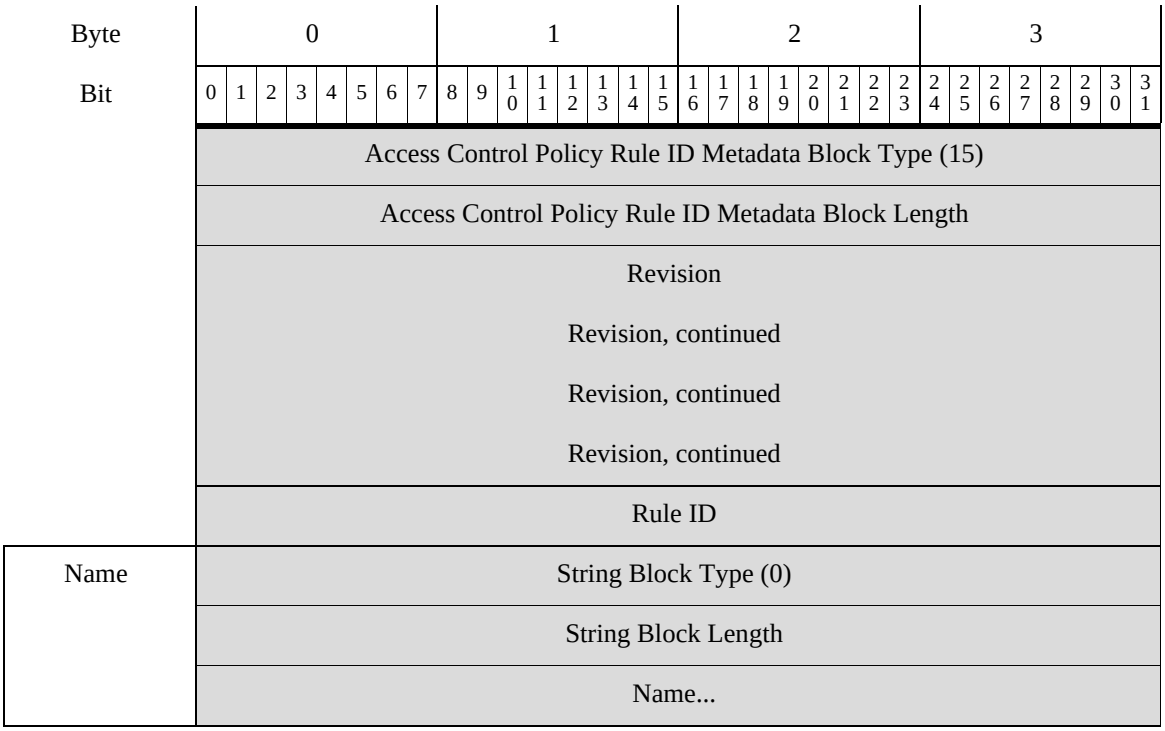
Table 3-30 Name Description Mapping Data Block Fields

Field	Data Type	Description
Name Description Mapping Block Type	uint32	Initiates a Name Description Mapping block. This value is always `61`.
Name Description Mapping Block Length	uint32	Total number of bytes in the Name Description Mapping block, including eight bytes for the Name Description Mapping block type and length fields, plus the number of bytes of data that follows.
ID	unit32	The unique identifier for the event or other object the ID identifies. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name associated with the ID. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Name field.
Name	string	The name of the event or object.
String Block Type	uint32	Initiates a String data block containing the description associated with the ID. This value is always `0`.
String Block Length	uint32	The number of bytes included in the description String data block, including eight bytes for the block type and header fields plus the number of bytes in the Description field.
Description	string	A description of the object or event associated with the ID.

Access Control Policy Rule ID Metadata Block

The eStreamer service uses the Access Control Policy Rule ID metadata block to contain information about access control policy rule IDs. This data block has a block type of 15 in series 2.

The following diagram shows the structure of the Access Control Policy Rule ID metadata block.



The following table describes the fields in the Access Control Policy Rule ID Metadata block.

Table 3-31 Access Control Policy Rule ID Metadata Block Fields

Field	Data Type	Description
Access Control Policy Rule ID Metadata Block Type	uint32	Initiates a Access Control Policy Rule ID Metadata block. This value is always `15`.
Access Control Policy Rule ID Metadata Block Length	uint32	Total number of bytes in the Access Control Policy Rule ID block, including eight bytes for the Access Control Policy Rule ID metadata block type and length fields, plus the number of bytes of data that follows.
Revision	uint8[16]	Revision number of the rule associated with the triggered correlation event.
Rule ID	uint32	Internal identifier for the rule that triggered the event. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the descriptive name associated with the access control policy rule. This value is always `0`.

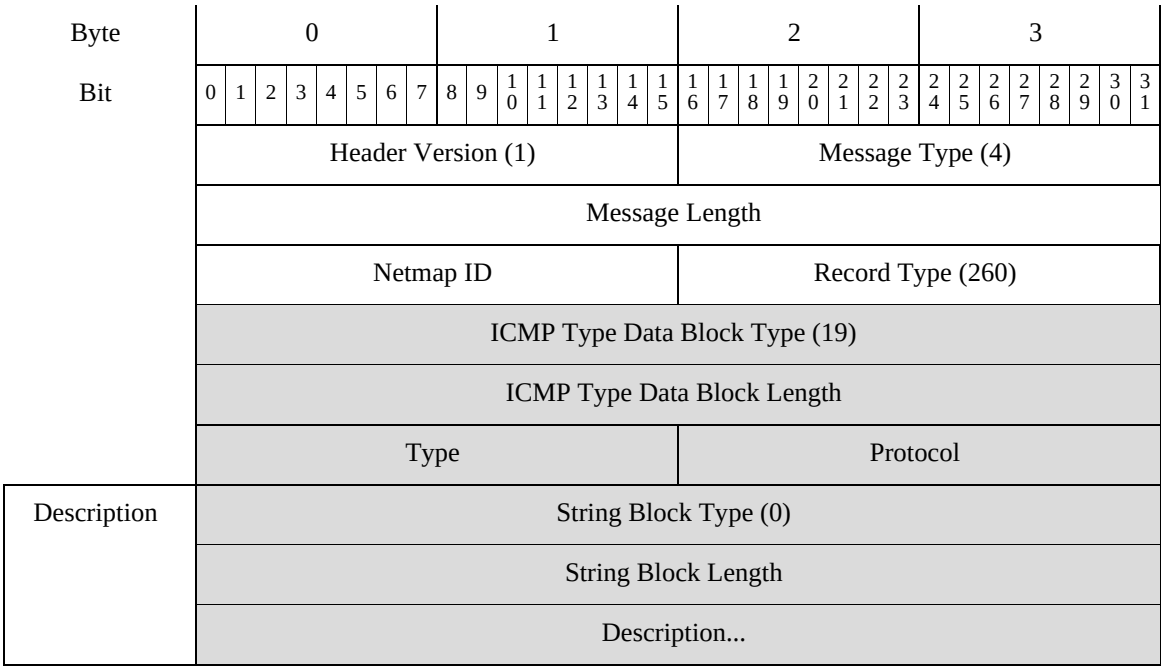
Table 3-31 Access Control Policy Rule ID Metadata Block Fields (continued)

Field	Data Type	Description
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Name field.
Name	string	The descriptive name of the access control policy rule.

ICMP Type Data Block

The eStreamer service uses the ICMP Type data block to contain information about ICMP Types. This data block has a record type of 260, and a block type of 19 in series 2.

The following diagram shows the structure of the ICMP Type data block.



The following table describes the fields in the ICMP Type data block.

Table 3-32 ICMP Type Data Block Fields

Field	Data Type	Description
ICMP Type Data Block Type	uint32	Initiates an ICMP Type data block. This value is always `19`.
ICMP Type Data Block Length	uint32	Total number of bytes in the ICMP Type data block, including eight bytes for the ICMP Type data block type and length fields, plus the number of bytes of data that follows.
Type	uint16	The ICMP type of the event.

Table 3-32 ICMP Type Data Block Fields (continued)

Field	Data Type	Description
Protocol	uint16	IANA-specified protocol number. For example: • 0 — IP • 1 — ICMP • 6 — TCP • 17 — UDP
String Block Type	uint32	Initiates a String data block containing the description of the ICMP type. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Description field.
Description	string	Description of the ICMP type for the event.

ICMP Code Data Block

The eStreamer service uses the ICMP Code data block to contain information about access control policy rule IDs. This data block has a record type of 270, and block type of 20 in series 2.

The following diagram shows the structure of the Access Control Policy Rule ID metadata block.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Header Version (1)																Message Type (4)															
	Message Length																															
	Netmap ID																Record Type (270)															
	ICMP Code Data Block Type (20)																															
	ICMP Code Data Block Length																															
	Code																Type															
Description	Protocol																String Block Type (0)															
	String Block Type (0), continued																String Block Length															
	String Block Length, continued																Description...															

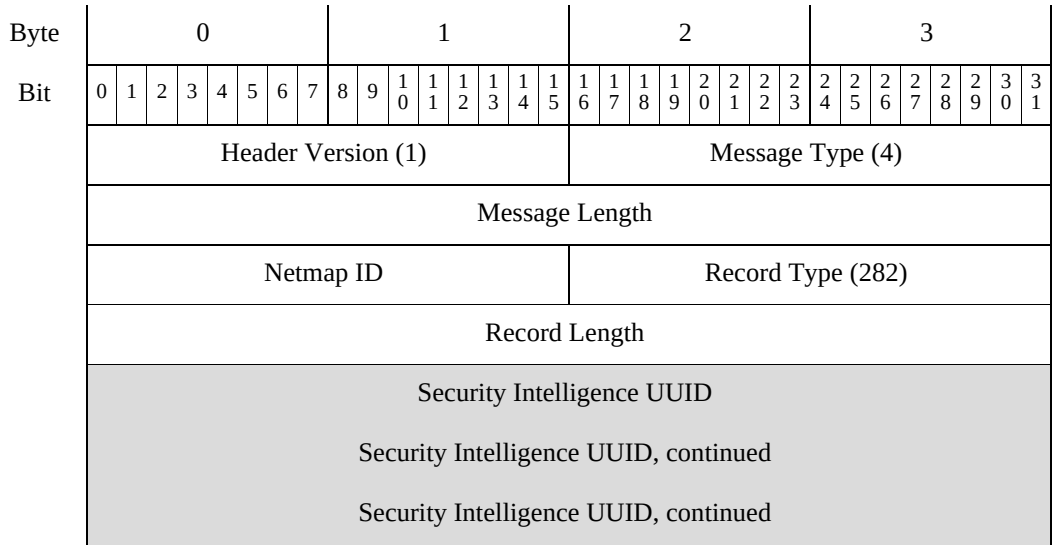
The following table describes the fields in the ICMP Code data block.

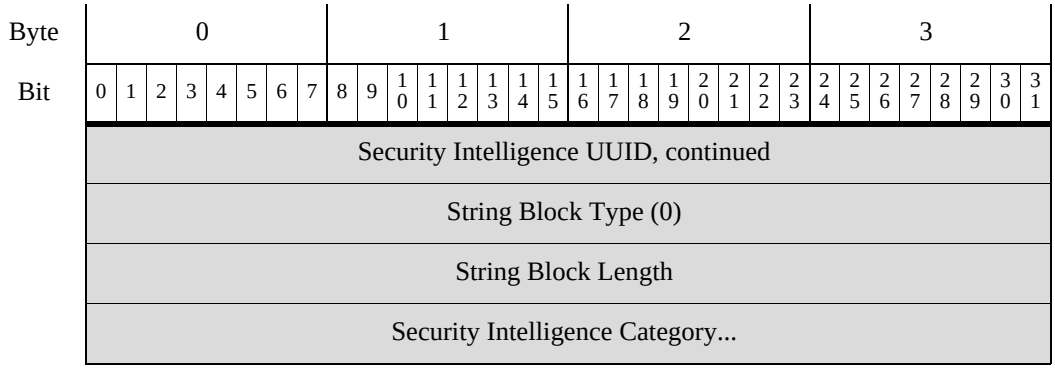
Table 3-33 ICMP Code Data Block Fields

Field	Data Type	Description
ICMP Code Data Block Type	uint32	Initiates a ICMP Code data block. This value is always 20.
ICMP Code Data Block Length	uint32	Total number of bytes in the ICMP Code data block, including eight bytes for the ICMP Code data block type and length fields, plus the number of bytes of data that follows.
Code	uint16	The ICMP code of the event.
Type	uint16	The ICMP type of the event.
Protocol	uint16	IANA-specified protocol number. For example: • 0 — IP • 1 — ICMP • 6 — TCP • 17 — UDP
String Block Type	uint32	Initiates a String data block containing the description of the ICMP code. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Description field.
Description	string	Description of the ICMP code for the event.

Security Intelligence Category Metadata for 5.4.1+

The eStreamer service transmits metadata containing Security Intelligence Category information, the format of which is shown below. Note that the Record Type field, which appears after the Message Length field, has a value of 282, indicating a Security Intelligence Category record.





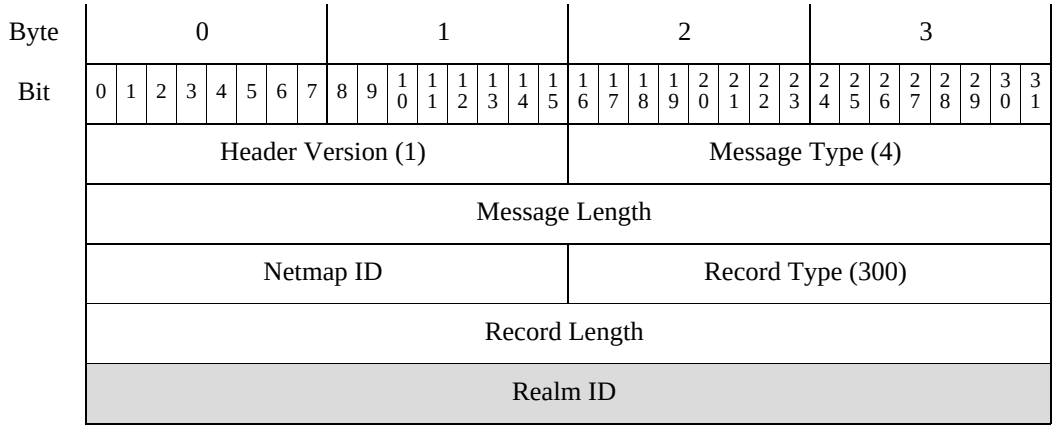
The following table describes the fields in the Security Context Name record.

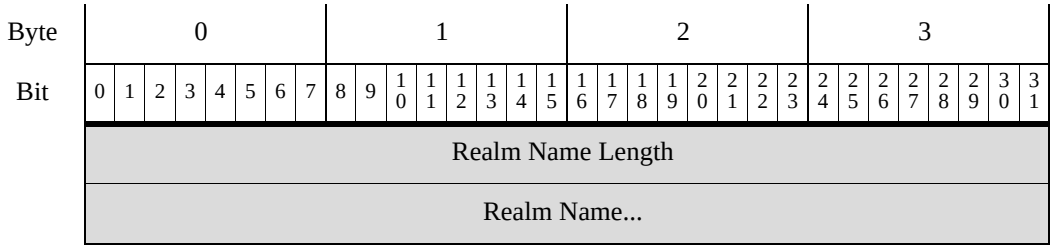
Table 3-34 Security Context Name Record Fields

Field	Data Type	Description
Security Intelligence UUID	uint8[16]	The UUID of the Security Intelligence. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the Security Intelligence category. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Security Intelligence Category String data block, including eight bytes for the block type and header fields plus the number of bytes in the Profile Name field.
Security Intelligence Category	string	The Security Intelligence Category.

Realm Metadata for 6.0+

The eStreamer service transmits metadata containing Realm information, the format of which is shown below. Note that the Record Type field, which appears after the Message Length field, has a value of `300`, indicating a Realm Metadata record.





The following table describes the fields in the Realm Metadata record.

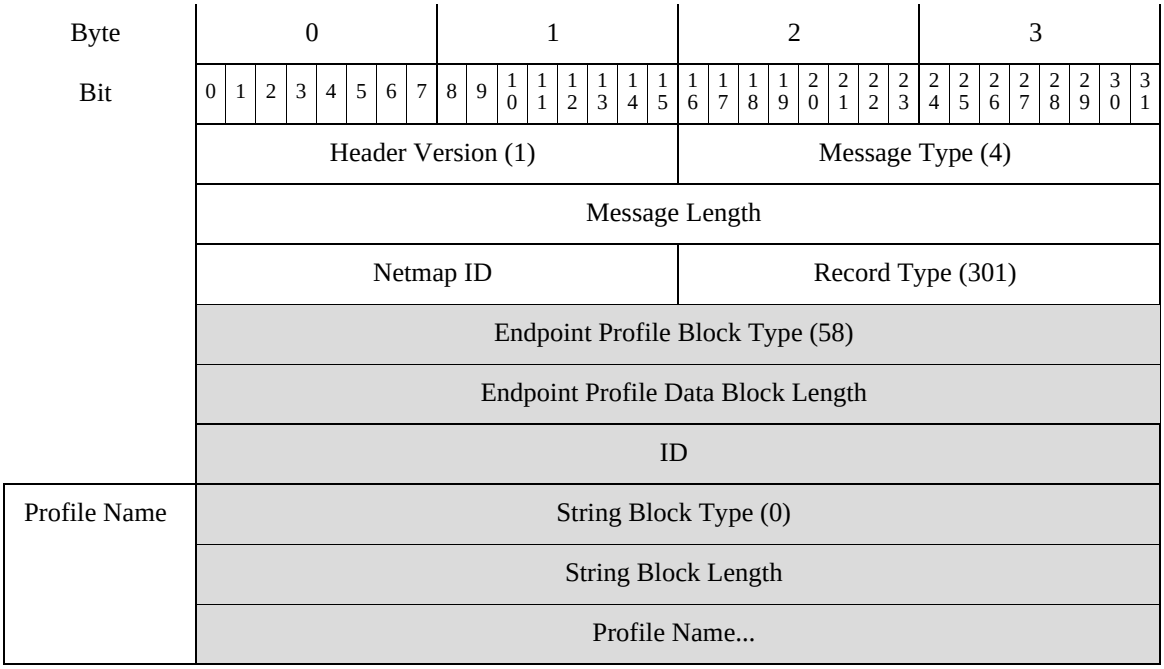
Table 3-35 Realm Metadata Record Fields

Field	Data Type	Description
Realm ID	uint32	The ID number of the realm. This field is the unique key for this record.
Realm Name Length	uint32	The number of bytes included in the Realm Name.
Realm Name	string	The realm name

Endpoint Profile Data Block for 6.0+

The eStreamer service uses the Endpoint Profile data block to contain information about network endpoints. This data block has a record type of 301, and block type of 58 in series 2.

The following diagram shows the structure of the Access Control Policy Rule ID metadata block.



Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Full Name	String Block Type (0)																															
	String Block Length																															
	Full Name...																															

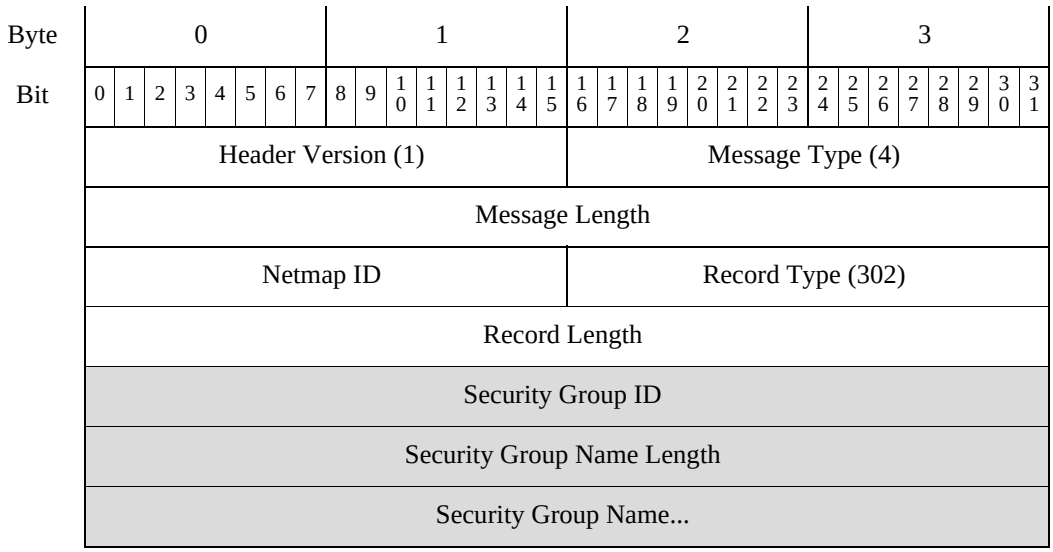
The following table describes the fields in the Endpoint Profile data block.

Table 3-36 Endpoint Profile Data Block Fields

Field	Data Type	Description
Endpoint Profile Data Block Type	uint32	Initiates a Endpoint Profile data block. This value is always `58`.
Endpoint Profile Data Block Length	uint32	Total number of bytes in the Endpoint Profile data block, including eight bytes for the Endpoint Profile data block type and length fields, plus the number of bytes of data that follows.
ID	uint32	ID number of the endpoint.
String Block Type	uint32	Initiates a String data block containing the profile of the endpoint. This value is always `0`.
String Block Length	uint32	The number of bytes included in the profile name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Profile Name field.
Profile Name	string	Name of the endpoint profile.
String Block Type	uint32	Initiates a String data block containing the full name of the endpoint. This value is always `0`.
String Block Length	uint32	The number of bytes included in the full name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Full Name field.
Full Name	string	Fully qualified name of the profile, providing the relationship hierarchy of the type of endpoint.

Security Group Metadata for 6.0+

The eStreamer service transmits metadata containing Security Group information, the format of which is shown below. Note that the Record Type field, which appears after the Message Length field, has a value of `302`, indicating a Security Group Metadata record.



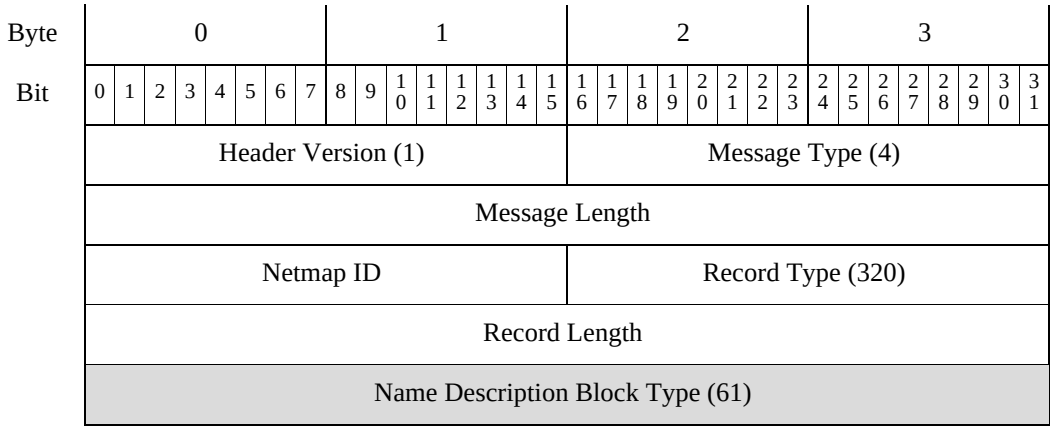
The following table describes the fields in the Security Group Metadata record.

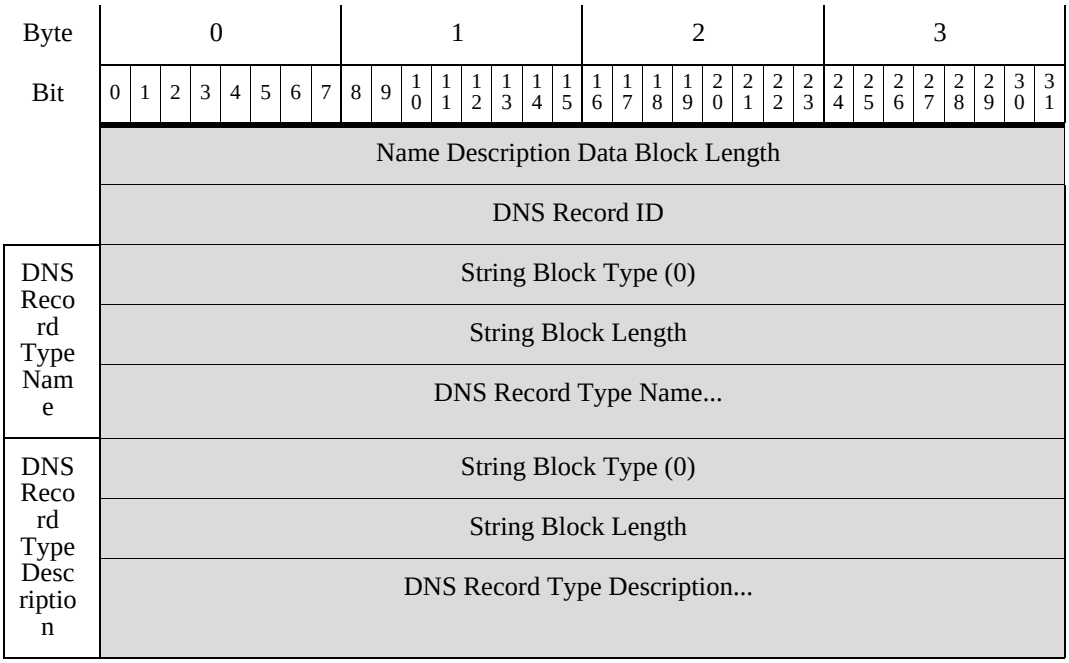
Table 3-37 Security Group Metadata Record Fields

Field	Data Type	Description
Security Group ID	uint32	The ID number of the security group. This field is the unique key for this record.
Security Group Name Length	uint32	The number of bytes included in the Security Group Name.
Security Group Name	string	The security group name

DNS Record Type Metadata for 6.0+

The eStreamer service transmits metadata containing DNS Record Type information, the format of which is shown below. Note that the Record Type field, which appears after the Message Length field, has a value of `320`, indicating a DNS Record Type Metadata record.





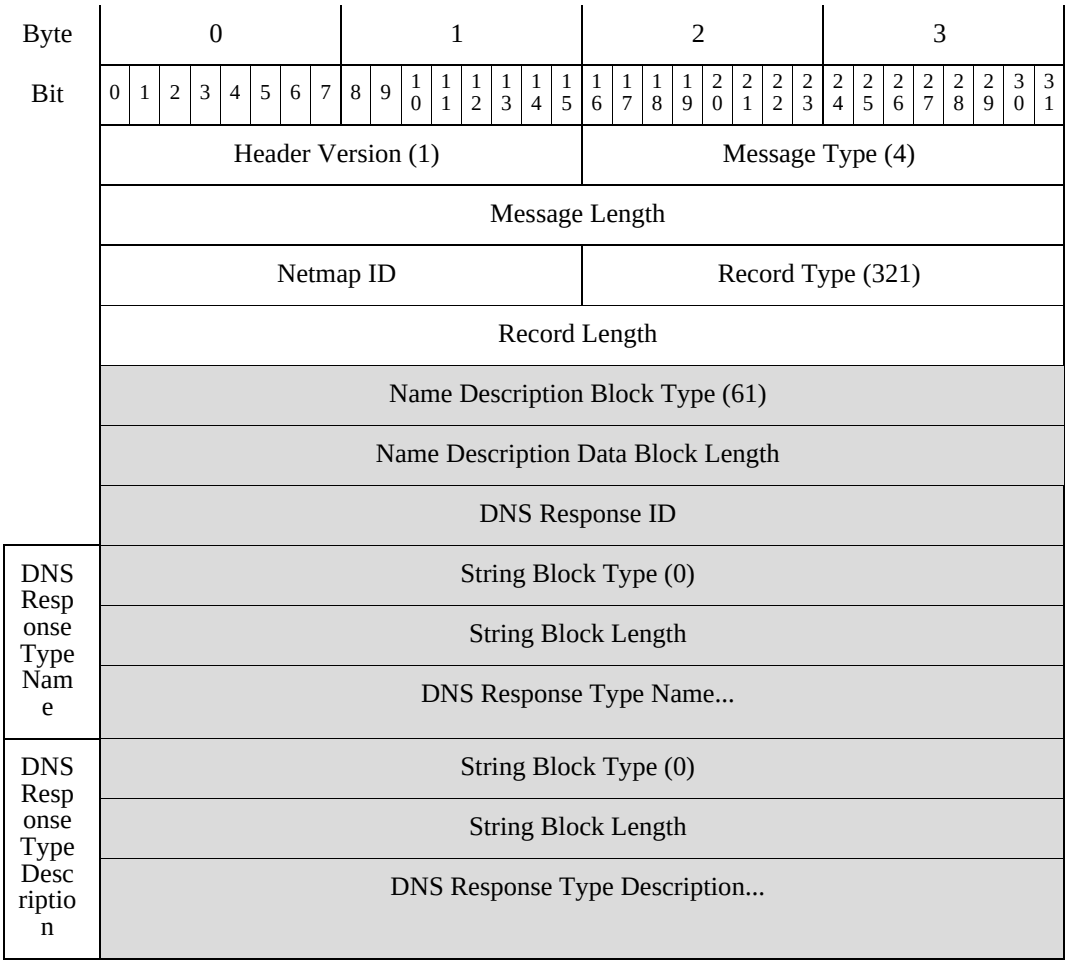
The following table describes the fields in the DNS Record Type Metadata record.

Table 3-38 DNS Record Type Metadata Fields

Field	Data Type	Description
Name Description Data Block Type	uint32	Initiates a Name Description data block. This value is always `61`.
Name Description Data Block Length	uint32	Total number of bytes in the Name Description data block, including eight bytes for the Name Description data block type and length fields, plus the number of bytes of data that follows.
DNS Record ID	uint32	The ID Number of the DNS Record. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the DNS Record Type. This value is always `0`.
String Block Length	uint32	The number of bytes included in the DNS Record Type Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the DNS Record Type Name field.
DNS Record Type Name	string	Name of the DNS Record Type.
String Block Type	uint32	Initiates a String data block containing the description of the DNS Record Type. This value is always `0`.
String Block Length	uint32	The number of bytes included in the DNS Record Type Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the DNS Record Type Description field.
DNS Record Type Description	string	Description of the DNS Record Type.

DNS Response Type Metadata for 6.0+

The eStreamer service transmits the DNS Response Type Metadata , the format of which is shown below. Note that the Record Type field, which appears after the Message Length field, has a value of `321`, indicating a DNS Response Type Metadata record.



The following table describes the fields in the DNS Response Type Metadata record.

Table 3-39 DNS Response Type Metadata Fields

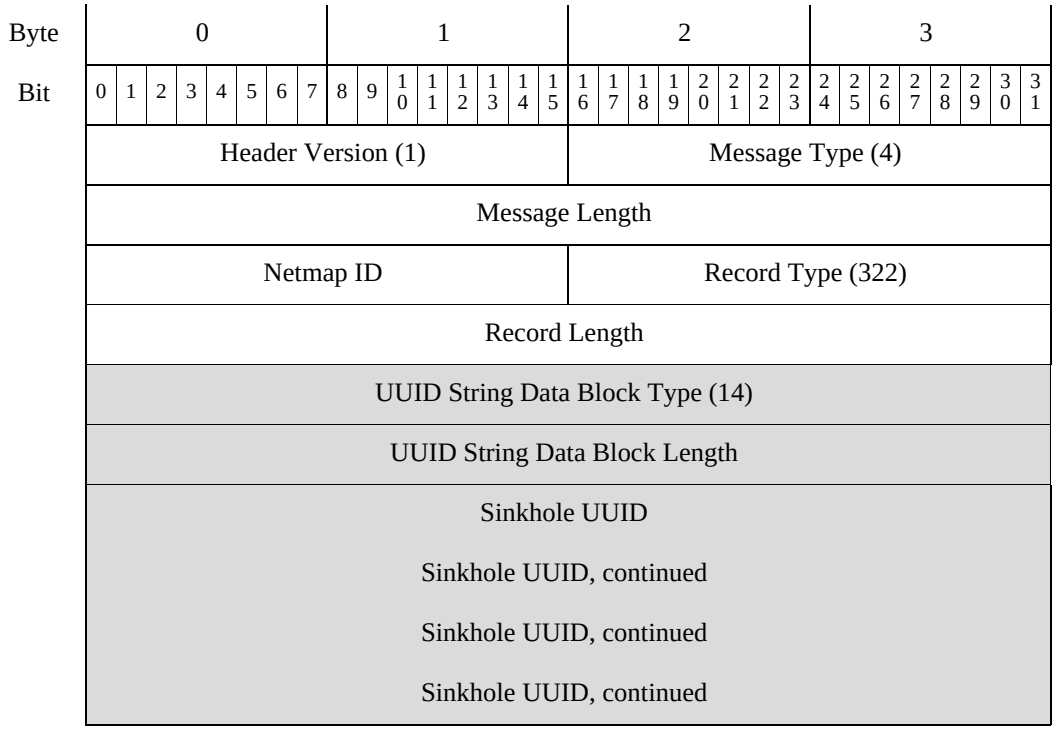
Field	Data Type	Description
Name Description Data Block Type	uint32	Initiates a Name Description data block. This value is always `61`.
Name Description Data Block Length	uint32	Total number of bytes in the Name Description data block, including eight bytes for the Name Description data block type and length fields, plus the number of bytes of data that follows.
DNS Response ID	uint32	The ID Number of the DNS Response. This field is the unique key for this record.

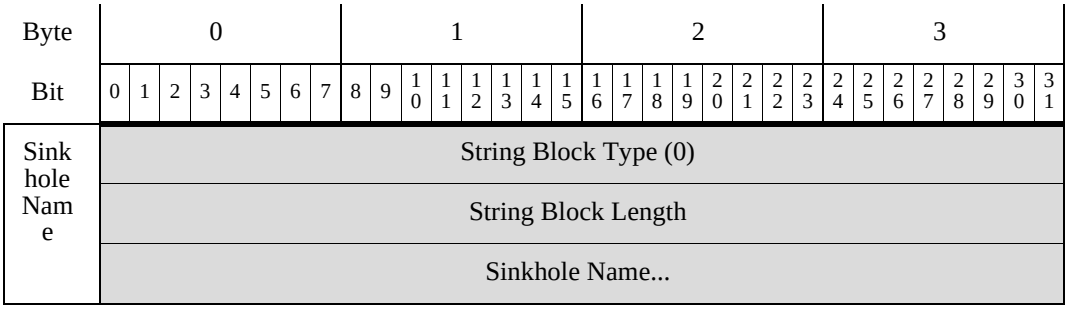
Table 3-39 DNS Response Type Metadata Fields (continued)

Field	Data Type	Description
String Block Type	uint32	Initiates a String data block containing the name of the DNS Response Type. This value is always `0`.
String Block Length	uint32	The number of bytes included in the DNS Response Type Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the DNS Response Type Name field.
DNS Response Type Name	string	Name of the DNS Response Type.
String Block Type	uint32	Initiates a String data block containing the description of the DNS Response Type. This value is always `0`.
String Block Length	uint32	The number of bytes included in the DNS Response Type Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the DNS Response Type Description field.
DNS Response Type Description	string	Description of the DNS Response Type.

Sinkhole Metadata for 6.0+

The eStreamer service transmits metadata containing Sinkhole information, the format of which is shown below. Note that the Record Type field, which appears after the Message Length field, has a value of `322`, indicating a Sinkhole Metadata record.





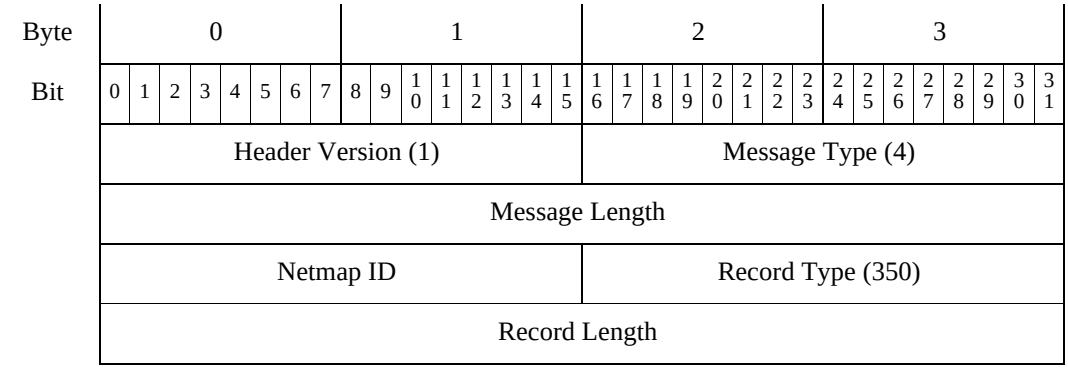
The following table describes the fields in the Sinkhole Metadata record.

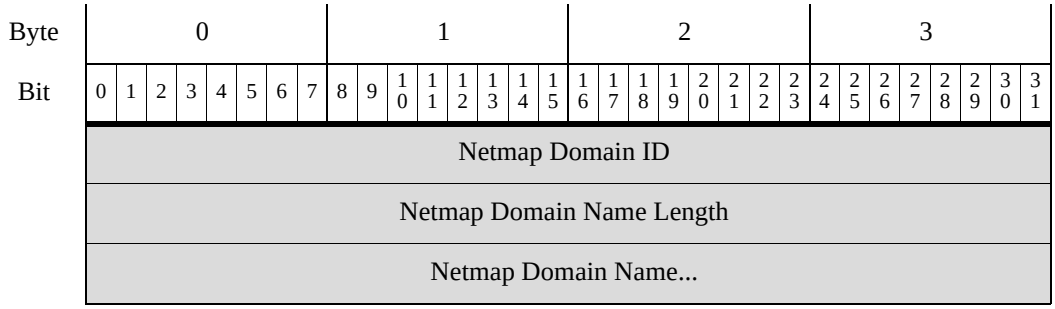
Table 3-40 Sinkhole Metadata Record Fields

Field	Data Type	Description
UUID String Data Block Type	uint32	Initiates a UUID String data block. This value is always `14`.
UUID String Data Block Length	uint32	Total number of bytes in the UUID String data block, including eight bytes for the UUID String data block type and length fields, plus the number of bytes of data that follows.
Sinkhole UUID	uint8[16]	The UUID number of the sinkhole. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the sinkhole. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Sinkhole Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Sinkhole Name field.
Sinkhole Name	string	Name of the Sinkholee.

Netmap Domain Metadata for 6.0+

The eStreamer service transmits metadata containing Netmap Domain information, the format of which is shown below. Note that the Record Type field, which appears after the Message Length field, has a value of `350`, indicating a Netmap Domain Metadata record.





The following table describes the fields in the Netmap Domain Metadata record.

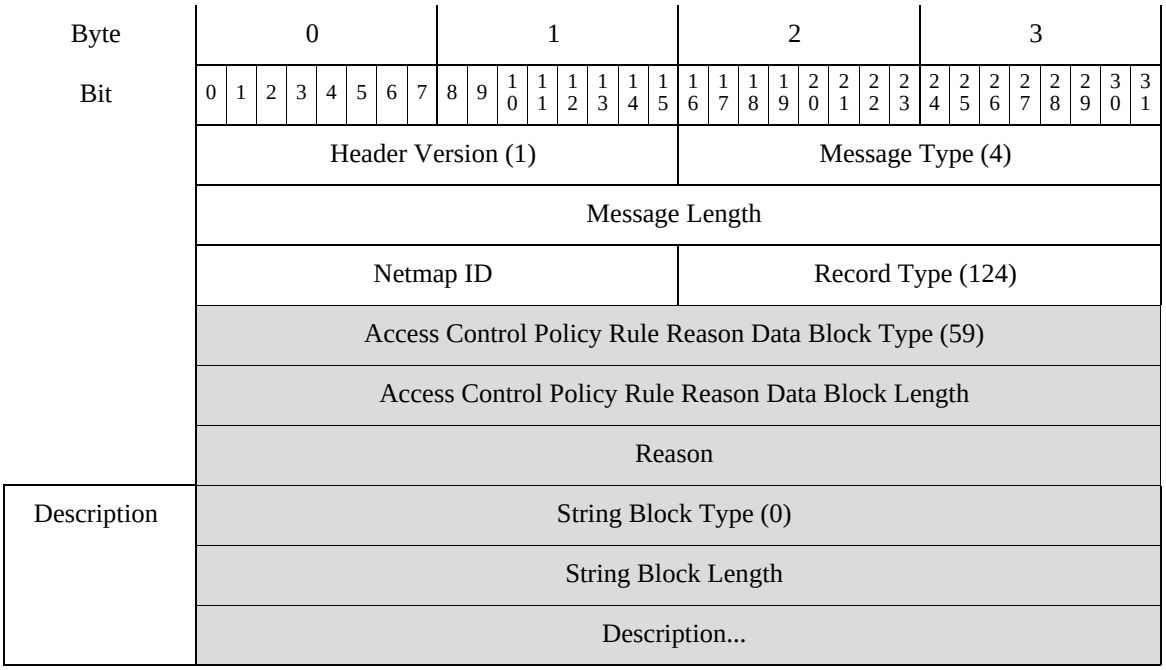
Table 3-41 Sinkhole Metadata Record Fields

Field	Data Type	Description
Netmap Domain ID	uint32	The ID number of the netmap domain. This field is the unique key for this record.
Netmap Domain Name Length	uint32	The number of bytes included in the Netmap Domain Name.
Netmap Domain Name	string	The netmap domain name

Access Control Policy Rule Reason Data Block for 6.0+

The eStreamer service uses the Access Control Rule Policy Rule Reason Data block to contain information about access control policy rules. This data block has a record type of 124, and a block type of 59 in series 2. It supersedes block type 21. The Reason field has been increased from 16 bits to 32 bits.

The following diagram shows the structure of the Access Control Policy Rule ID metadata block.



The following table describes the fields in the Access Control Policy Rule Reason data block.

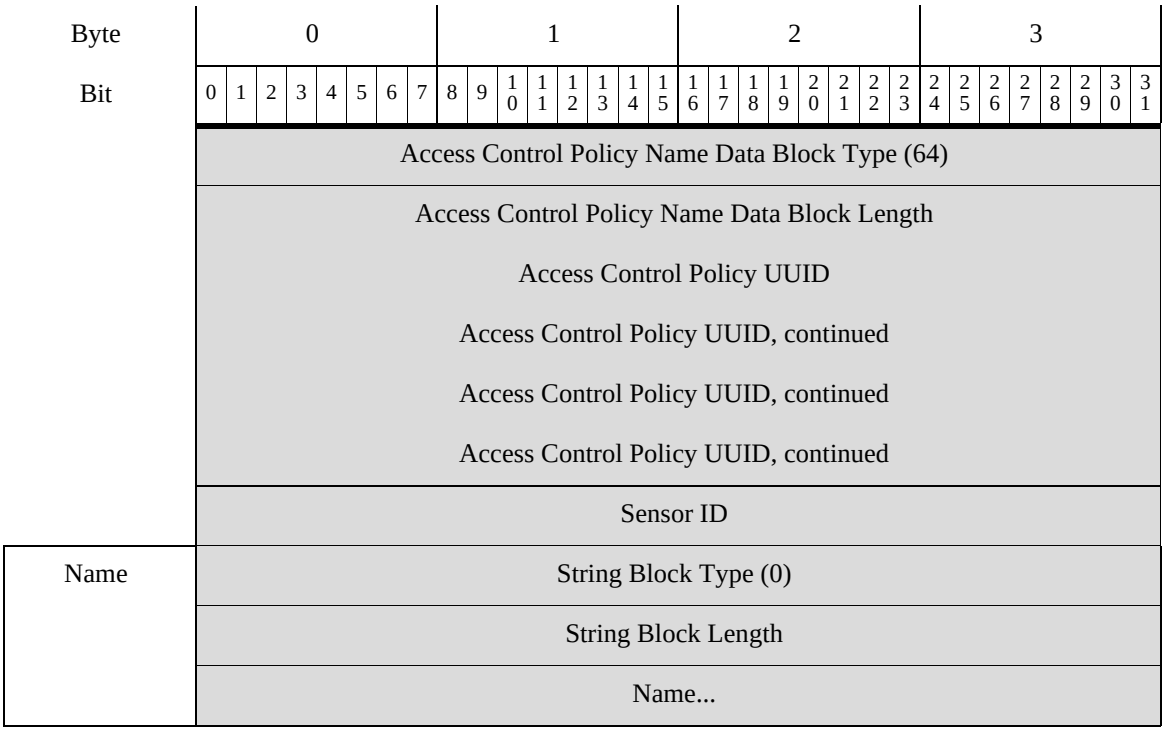
Table 3-42 Access Control Policy Rule Reason Data Block Fields

Field	Data Type	Description
Access Control Policy Rule Reason Data Block Type	uint32	Initiates an Access Control Policy Rule Reason data block. This value is always `59`.
Access Control Policy Rule Reason Data Block Length	uint32	Total number of bytes in the Access Control Policy Rule Reason data block, including eight bytes for the Access Control Policy Rule Reason data block type and length fields, plus the number of bytes of data that follows.
Reason	uint32	The number of the reason for the rule that triggered the event. Rule reasons are a binary bitmap in which multiple bits may be set. There may be several reasons for a rule. The bit values are as follows: • **1** — IP Block • **2** — IP Monitor • **4** — User Bypass • **8** — File Monitor • **16** — File Block • **32** — Intrusion Monitor • **64** — Intrusion Block • **128** — File Resume Block • **256** — File Resume Allow"] • **512** — File Custom Detection • **1024** — SSL Block • **2048** — DNS Block • **4096** — DNS Monitor • **8192** — URL Block • **16384** — URL Monitor • **32768** — Content Restriction • **65536** — Intelligent App Bypass • **131072** — WSA Threat
String Block Type	uint32	Initiates a String data block containing the description of the access control policy rule reason. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Description field.
Description	string	Description of the reason for the rule.

Access Control Policy Name Data Block

The eStreamer service uses the Access Control Policy Name Data block to contain information about access control policy names. This data block has a a block type of 64 in series 2.

The following diagram shows the structure of the Access Control Policy Name metadata block.



The following table describes the fields in the Access Control Policy Name metadata block.

Table 3-43 Access Control Policy Policy Name Data Block Fields

Field	Data Type	Description
Access Control Policy Name Data Block Type	uint32	Initiates an Access Control Policy Name data block. This value is always `64`.
Access Control Policy Name Data Block Length	uint32	Total number of bytes in the Access Control Policy Name data block, including eight bytes for the Access Control Policy Name data block type and length fields, plus the number of bytes of data that follows.
Access Control Policy UUID	uint8[16]	UUID of the Access Control Policy
Sensor ID	uint32	ID Number of the sensor associated with the access control policy
String Block Type	uint32	Initiates a String data block containing the name of the access control policy. This value is always `0`.

Table 3-43 Access Control Policy Policy Name Data Block Fields (continued)

Field	Data Type	Description
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Name field.
Name	string	Name of the access control policy

IP Reputation Category Data Block

The eStreamer service uses the IP Reputation Category Data block to contain information about rule reputation categories. This data block has a block type of 22 in series 2.

The following diagram shows the structure of the IP Reputation Category data block.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	IP Reputation Category Data Block Type (22)																															
	IP Reputation Category Data Block Length																															
	Rule ID																															
	Policy UUID Policy UUID, continued Policy UUID, continued Policy UUID, continued																															
Description	String Block Type (0)																															
	String Block Length																															
	Category Name...																															

The following table describes the fields in the IP Reputation Category Data Block.

Table 3-44 IP Reputation Category Data Block Fields

Field	Data Type	Description
IP Reputation Category Data Block Type	uint32	Initiates a IP Reputation Category data block. This value is always `22`.
IP Reputation Category Data Block Length	uint32	Total number of bytes in the IP Reputation Category data block, including eight bytes for the IP Reputation Category data block type and length fields, plus the number of bytes of data that follows.

Table 3-44 IP Reputation Category Data Block Fields (continued)

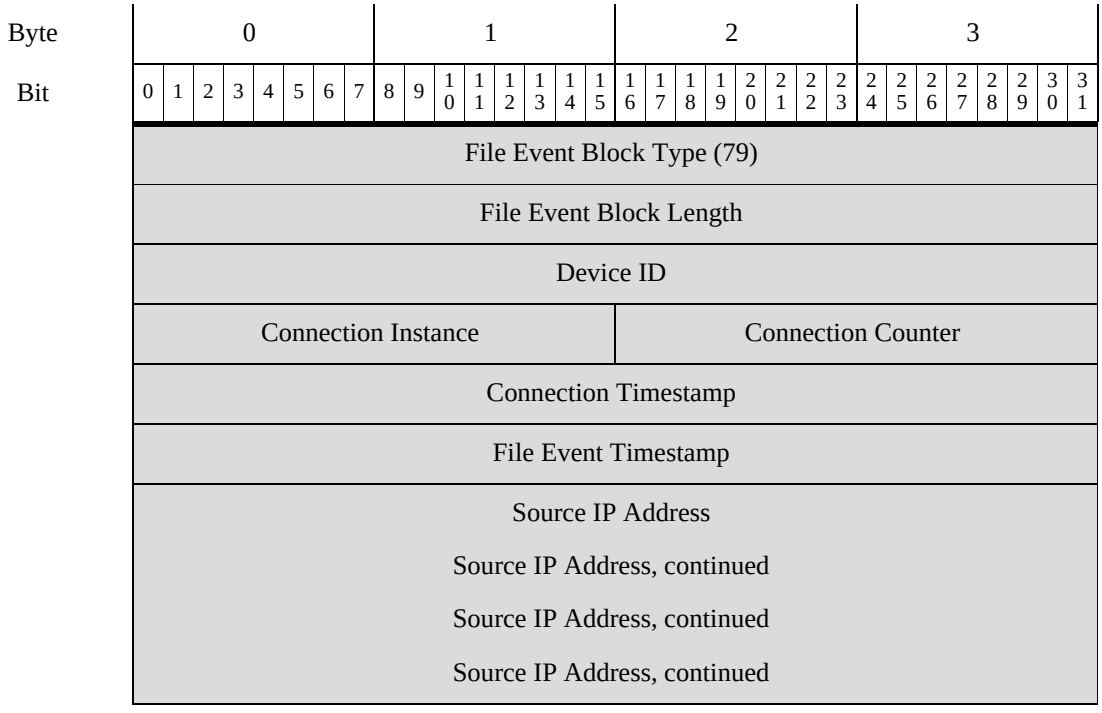
Field	Data Type	Description
Rule ID	uint32	Internal identifier for the rule that triggered the event.
Policy UUID	uint8[16]	UUID of the policy that triggered the event.
String Block Type	uint32	Initiates a String data block containing the description of the IP Reputation Category. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Category Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Category Name field.
Category Name	string	Name of the category for the rule.

File Event for 7.0+

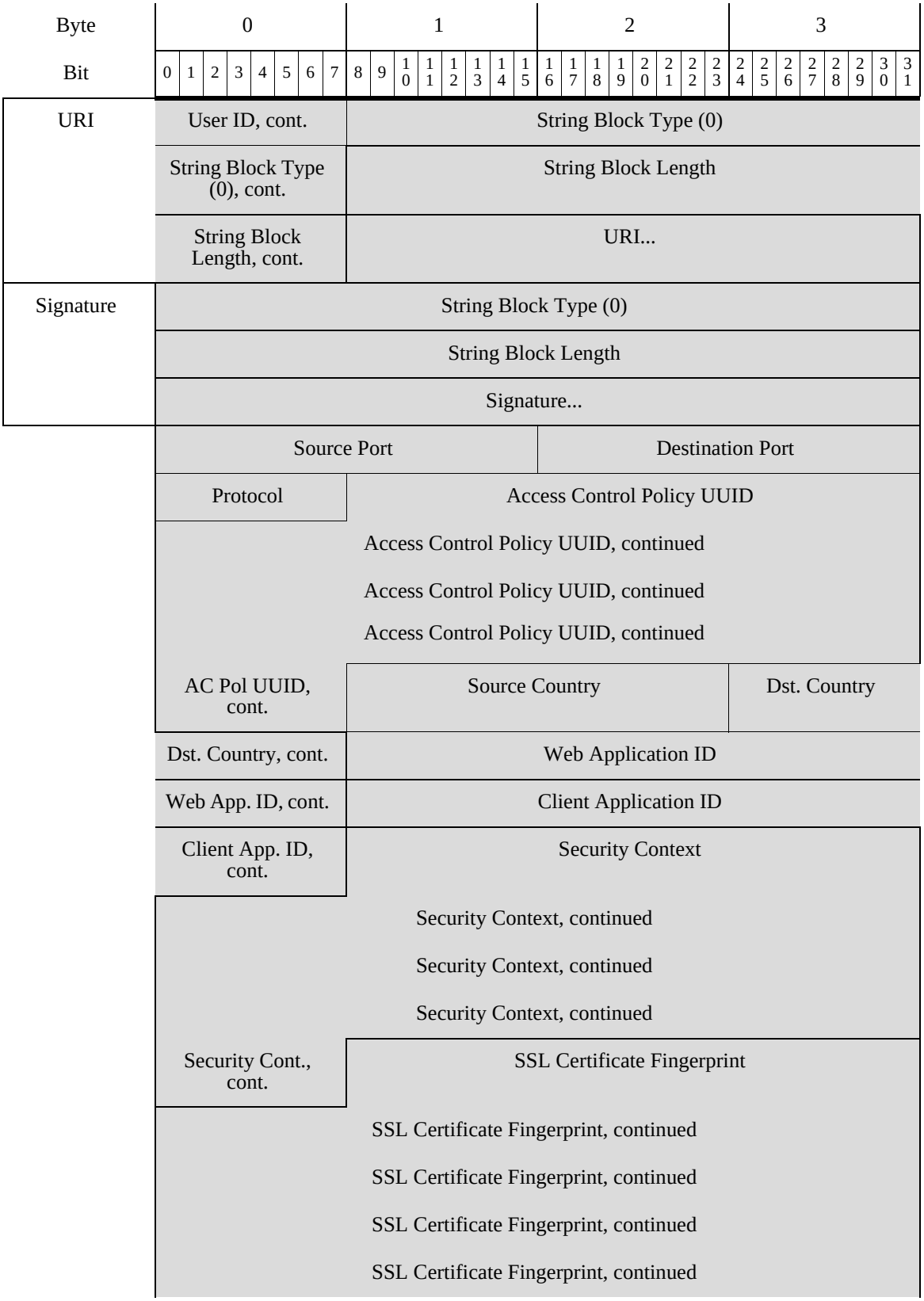
The File Event data block contains information on files that are sent over the network. This includes the connection information, whether the file is malware, and specific information to identify the file. The file event has a block type of 79 in the series 2 group of blocks. It supersedes block type 56. Fields for virtual routing and forwarding.

You request file event records by setting the file event flag—bit 30 in the Request Flags field—in the request message with an event version of 7and an event code of 111. See Request Flags, page 2-13. If you enable bit 23, an extended event header is included in the record.

The following graphic shows the structure of the File Event data block.



Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Destination IP Address Destination IP Address, continued Destination IP Address, continued Destination IP Address, continued																															
	Disposition								SPERO Disposition								File Storage Status								File Analysis Status							
	Local Malware Analysis Stat.								Archive File Status								Threat Score								Action							
	SHA Hash SHA Hash, continued SHA Hash, continued SHA Hash, continued SHA Hash, continued SHA Hash, continued SHA Hash, continued SHA Hash, continued																															
	File Type ID																															
File Name	String Block Type (0)																															
	String Block Length																															
	File Name...																															
	File Size File Size, continued																															
	Direction								Application ID																							
	App ID, cont.								User ID																							


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
URI
User ID, cont.
String Block Type (0)
String Block Type (0), cont.
String Block Length
String Block Length, cont.
URI...
Signature
String Block Type (0)
String Block Length
Signature...
Source Port
Destination Port
Protocol
Access Control Policy UUID
Access Control Policy UUID, continued
Access Control Policy UUID, continued
Access Control Policy UUID, continued
AC Pol UUID, cont.
Source Country
Dst. Country
Dst. Country, cont.
Web Application ID
Web App. ID, cont.
Client Application ID
Client App. ID, cont.
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
Security Cont., cont.
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	SSL Cert. Fpt., cont.								SSL Actual Action																SSL Flow Status							
Archive SHA	SSL Flow Stat., cont.								String Block Type (0)																							
	Str. Blk Type, cont.								String Length																							
	Str. Length, cont.								Archive SHA...																							
Archive Name	String Block Type (0)																															
	String Block Length																															
	Archive Name...																															
	Archive Depth								HTTP Response Code																							
Ingress VRF	HTTP Rsp Code, cont.								String Block Type (0)																							
	Str. Blk Type (0), cont.								String Block Length																							
	Str. Block Lgth, cont.								Ingress VRF Name...																							
Egress VRF	String Block Type (0)																															
	String Block Length																															
	Egress VRF Name...																															

The following table describes the fields in the file event data block.

Table 3-45 File Event Data Block for 7.0+ Fields

Field	Data Type	Description
File Event Block Type	uint32	Initiates whether file event data block. This value is always 79.
File Event Block Length	uint32	Total number of bytes in the file event block, including eight bytes for the file event block type and length fields, plus the number of bytes of data that follows.
Device ID	uint32	ID for the device that generated the event.
Connection Instance	uint16	Snort instance on the device that generated the event. Used to link the event with a connection or intrusion event.

Table 3-45 File Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
Connection Counter	uint16	Value used to distinguish between connection events that happen during the same second.
Connection Timestamp	uint32	UNIX timestamp (seconds since 01/01/1970) of the associated connection event.
File Event Timestamp	uint32	UNIX timestamp (seconds since 01/01/1970) of when the file type is identified and the file event generated.
Source IP Address	uint8[16]	IPv4 or IPv6 address for the source of the connection.
Destination IP Address	uint8[16]	IPv4 or IPv6 address for the destination of the connection.
Disposition	uint8	The malware status of the file. Possible values include: • 1 — CLEAN The file is clean and does not contain malware. • 2 — UNKNOWN It is unknown whether the file contains malware. • 3 — MALWARE The file contains malware. • 4 — UNAVAILABLE The software was unable to send a request to the AMP cloud for a disposition, or the AMP cloud services did not respond to the request. • 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user.
SPERO Disposition	uint8	Indicates whether the SPERO signature was used in file analysis. If the value is 1, 2, or 3, SPERO analysis was used. If there is any other value SPERO analysis was not used.
File Storage Status	uint8	The storage status of the file. Possible values are: • 1 — File Stored • 2 — File Stored • 3 — Unable to Store File • 4 — Unable to Store File • 5 — Unable to Store File • 6 — Unable to Store File • 7 — Unable to Store File • 8 — File Size is Too Large • 9 — File Size is Too Small • 10 — Unable to Store File • 11 — File Not Stored, Disposition Unavailable

Table 3-45 File Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
File Analysis Status	uint8	Indicates whether the file was sent for dynamic analysis. Possible values are: • 0 — File Not Sent for Analysis • 1 — Sent for Analysis • 2 — Sent for Analysis • 4 — Sent for Analysis • 5 — Failed to Send • 6 — Failed to Send • 7 — Failed to Send • 8 — Failed to Send • 9 — File Size is Too Small • 10 — File Size is Too Large • 11 — Sent for Analysis • 12 — Analysis Complete • 13 — Failure (Network Issue) • 14 — Failure (Rate Limit) • 15 — Failure (File Too Large) • 16 — Failure (File Read Error) • 17 — Failure (Internal Library Error) • 19 — File Not Sent, Disposition Unavailable • 20 — Failure (Cannot Run File) • 21 — Failure (Analysis Timeout) • 22 — Sent for Analysis • 23 —File Transmit File Capacity Handled — File capacity handled (stored on the sensor) because file could not be submitted to the sandbox for analysis • 25 — File Transmit Server Limited Exceeded Capacity Handled — File capacity handled due to rate limiting on server • 26 — Communication Failure — File capacity handled due to cloud connectivity failure • 27 — Not Sent — File not sent due to configuration • 28 — Preclass No Match —File not sent for dynamic analysis since pre-classification didn't find any embedded or suspicious object in the file • 29 — Transmit Sent Sandbox Private Cloud — File sent to the private cloud for dynamic analysis • 30 — Transmit Not Send Sendbox Private Cloud - File not sent to the private cloud for analysis

Table 3-45 File Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
Local Malware Analysis Status	uint8	The malware analysis status of the file. Possible values are: • 0 — File not Analyzed • 1 — Analysis Done • 2 — Analysis Failed • 3 — Manual Analysis Request
Archive File Status	uint8	The status of an archive being inspected. Can have the following values: • 0 — N/A — File is not being inspected as an archive • 1 — Pending — Archive is being inspected • 2 — Extracted — Successfully inspected without any problems • 3 — Failed — Failed to inspect, insufficient system resources • 4 — Depth Exceeded — Successful, but archive exceeded the nested inspection depth • 5 — Encrypted — Partially Successful, Archive was or contains an archive that is encrypted • 6 — Not Inspectable — Partially Successful, File is possibly Malformed or Corrupt
Threat Score	uint8	A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis.
Action	uint8	The action taken on the file based on the file type. Can have the following values: • 1 — Detect • 2 — Block • 3 — Malware Cloud Lookup • 4 — Malware Block • 5 — Malware Allow List • 6 — Cloud Lookup Timeout • 7 — Custom Detection • 8 — Custom Detection Block • 9 — Archive Block (Depth Exceeded) • 10 — Archive Block (Encrypted) • 11 — Archive Block (Failed to Inspect)
SHA Hash	uint8[32]	SHA-256 hash of the file, in binary format.

Table 3-45 File Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
File Type ID	uint32	ID number that maps to the file type. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information.
File Name	string	Name of the file.
File Size	uint64	Size of the file in bytes.
Direction	uint8	Value that indicates whether the file was uploaded or downloaded. Can have the following values: • 1 — Download • 2 — Upload Currently the value depends on the protocol (for example, if the connection is HTTP it is a download).
Application ID	uint32	ID number that maps to the application using the file transfer.
User ID	uint32	ID number for the user logged into the destination host, as identified by the system.
URI	string	Uniform Resource Identifier (URI) of the connection.
Signature	string	SHA-256 hash of the file, in string format.
Source Port	uint16	Port number for the source of the connection.
Destination Port	uint16	Port number for the destination of the connection.
Protocol	uint8	IANA protocol number specified by the user. For example: • 1 — ICMP • 4 — IP • 6 — TCP • 17 — UDP This is currently only TCP.
Access Control Policy UUID	uint8[16]	Unique identifier for the access control policy that triggered the event.
Source Country	uint16	Code for the country of the source host.
Destination Country	uint16	Code for the country of the destination host.
Web Application ID	uint32	The internal identification number for the web application, if applicable.
Client Application ID	uint32	The internal identification number for the client application, if applicable.
Security Context	uint8(16)	ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode.

Table 3-45 File Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
SSL Certificate Fingerprint	uint8[20]	SHA1 hash of the SSL Server certificate.
SSL Actual Action	uint16	The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include: • 0 — 'Unknown' • 1 — 'Do Not Decrypt' • 2 — 'Block' • 3 — 'Block With Reset' • 4 — 'Decrypt (Known Key)' • 5 — 'Decrypt (Replace Key)' • 6 — 'Decrypt (Resign)'

Table 3-45 File Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
SSL Flow Status	uint16	Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include: • 0 — 'Unknown' • 1 — 'No Match' • 2 — 'Success' • 3 — 'Uncached Session' • 4 — 'Unknown Cipher Suite' • 5 — 'Unsupported Cipher Suite' • 6 — 'Unsupported SSL Version' • 7 — 'SSL Compression Used' • 8 — 'Session Undecryptable in Passive Mode' • 9 — 'Handshake Error' • 10 — 'Decryption Error' • 11 — 'Pending Server Name Category Lookup' • 12 — 'Pending Common Name Category Lookup' • 13 — 'Internal Error' • 14 — 'Network Parameters Unavailable' • 15 — 'Invalid Server Certificate Handle' • 16 — 'Server Certificate Fingerprint Unavailable' • 17 — 'Cannot Cache Subject DN' • 18 — 'Cannot Cache Issuer DN' • 19 — 'Unknown SSL Version' • 20 — 'External Certificate List Unavailable' • 21 — 'External Certificate Fingerprint Unavailable' • 22 — 'Internal Certificate List Invalid' • 23 — 'Internal Certificate List Unavailable' • 24 — 'Internal Certificate Unavailable' • 25 — 'Internal Certificate Fingerprint Unavailable' • 26 — 'Server Certificate Validation Unavailable' • 27 — 'Server Certificate Validation Failure' • 28 — 'Invalid Action'
String Block Type	uint32	Initiates a String data block containing the Archive SHA. This value is always 0.

Table 3-45 File Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
String Block Length	uint32	The number of bytes included in the Archive SHA String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name.
Archive SHA	string	SHA1 hash of the parent archive in which the file is contained.
String Block Type	uint32	Initiates a String data block containing the Archive Name. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Archive Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name.
Archive Name	string	Name of the parent archive.
Archive Depth	uint8	Number of layers in which the file is nested. For example, if a text file is in a zip archive, this has a value of `1`.
HTTP Response Code	uint32	HTTP Response Code.
String Block Type	uint32	Initiates a String data block containing the name of the ingress VRF. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Ingress VRF name field.
Ingress VRF Name	string	The virtual router through which traffic entered the network.
String Block Type	uint32	Initiates a String data block containing the name of the egress VRF. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Egress VRF name field.
Egress VRF Name	string	The name of the virtual router through which traffic exited the network.

Malware Event Data Block 7.0+

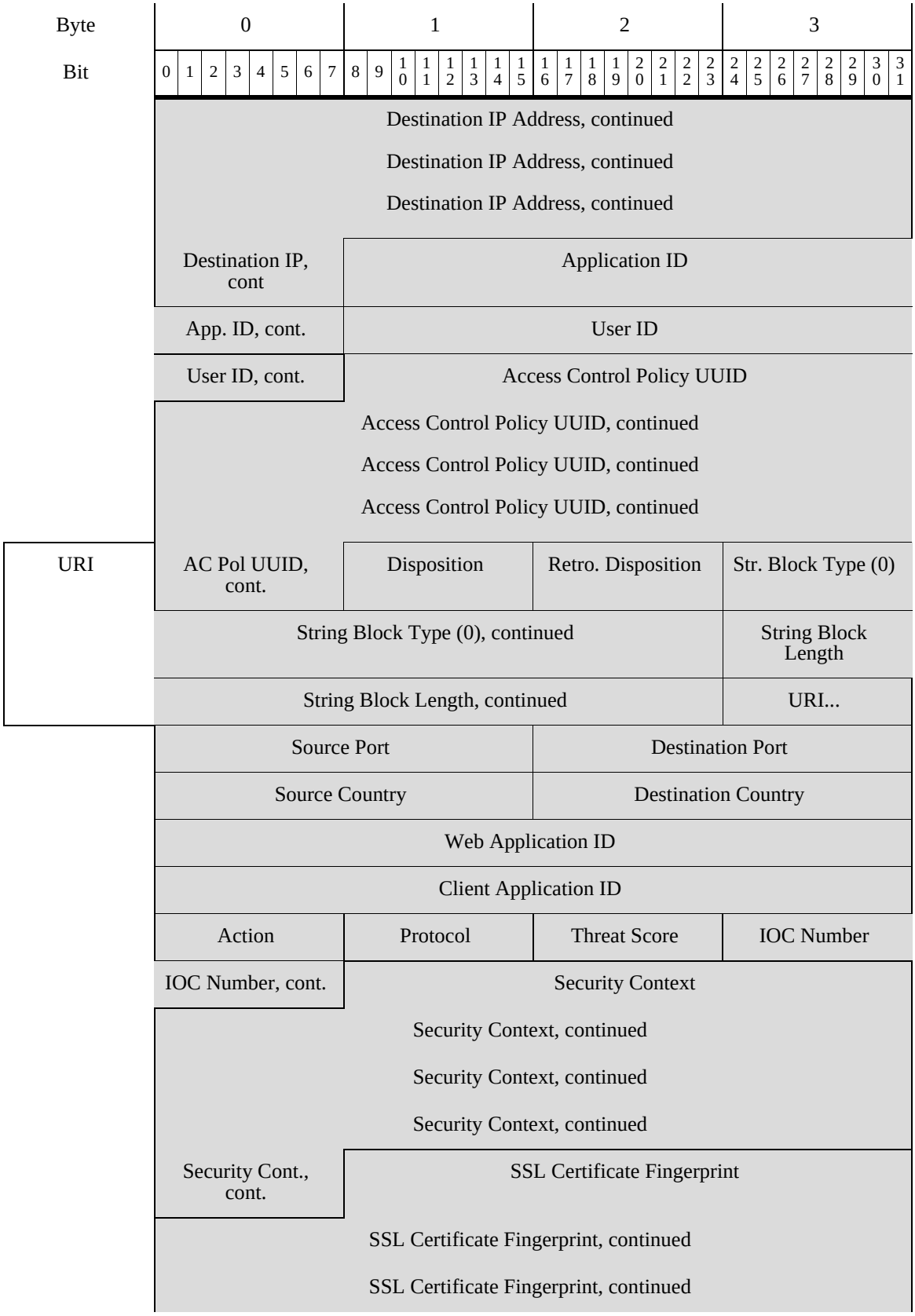
The eStreamer service uses the malware event data block to store information on malware events. These events contain information on malware detected or quarantined within a cloud, the detection method, and hosts and users affected by the malware. The malware event data block has a block type of 80 in the series 2 group of blocks. It supersedes block 62. Virtual routing and forwarding fields have been added.

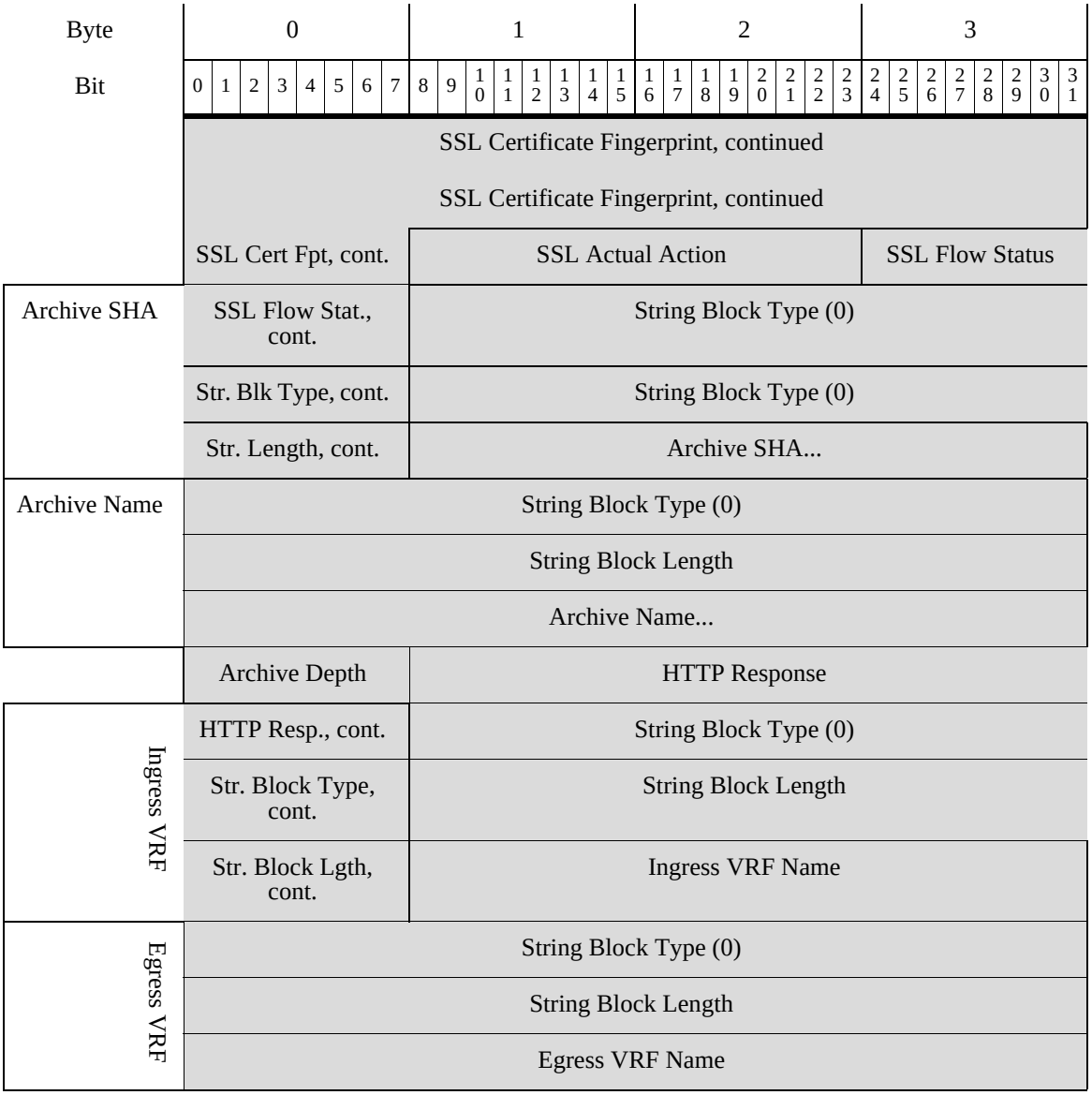
You request the event as part of the malware event record by setting the malware event flag—bit 30 in the request flags field—in the request message with an event version of 8 and an event code of 101.

The following graphic shows the structure of the malware event data block.

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Malware Event Block Type (80)																															
	Malware Event Block Length																															
	Agent UUID Agent UUID, continued Agent UUID, continued Agent UUID, continued																															
	Cloud UUID Cloud UUID, continued Cloud UUID, continued Cloud UUID, continued																															
	Malware Event Timestamp																															
	Event Type ID																															
	Event Subtype ID																															
Detection Name	Detector ID								String Block Type (0)																							
	String Block Type (0), cont.								String Block Length																							
	String Block Length, cont.								Detection Name...																							
User	String Block Type (0)																															
	String Block Length																															
	User...																															
File Name	String Block Type (0)																															
	String Block Length																															
	File Name...																															

Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
File Path
String Block Type (0)
String Block Length
File Path...
File SHA Hash
String Block Type (0)
String Block Length
File SHA Hash...
File Size
File Type
File Timestamp
Parent File Name
String Block Type (0)
String Block Length
Parent File Name...
Parent File SHA Hash
String Block Type (0)
String Block Length
Parent File SHA Hash...
Event Description
String Block Type (0)
String Block Length
Event Description...
Device ID
Connection Instance
Connection Counter
Connection Event Timestamp
Direction
Source IP Address
Source IP Address, continued
Source IP Address, continued
Source IP Address, continued
Source IP, cont.
Destination IP Address


Byte
0
1
2
3
Bit
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Destination IP Address, continued
Destination IP Address, continued
Destination IP Address, continued
Destination IP, cont
Application ID
App. ID, cont.
User ID
User ID, cont.
Access Control Policy UUID
Access Control Policy UUID, continued
Access Control Policy UUID, continued
Access Control Policy UUID, continued
URI
AC Pol UUID, cont.
Disposition
Retro. Disposition
Str. Block Type (0)
String Block Type (0), continued
String Block Length
String Block Length, continued
URI...
Source Port
Destination Port
Source Country
Destination Country
Web Application ID
Client Application ID
Action
Protocol
Threat Score
IOC Number
IOC Number, cont.
Security Context
Security Context, continued
Security Context, continued
Security Context, continued
Security Cont., cont.
SSL Certificate Fingerprint
SSL Certificate Fingerprint, continued
SSL Certificate Fingerprint, continued



The following table describes the fields in the malware event data block.

Table 3-46 Malware Event Data Block for 7.0+ Fields

Field	Data Type	Description
Malware Event Block Type	uint32	Initiates a malware event data block. This value is always 80.
Malware Event Block Length	uint32	Total number of bytes in the malware event data block, including eight bytes for the malware event block type and length fields, plus the number of bytes of data that follows.
Agent UUID	uint8[16]	The internal unique ID of the AMP for Endpoints agent reporting the malware event.
Cloud UUID	uint8[16]	The internal unique ID of the AMP cloud from which the malware event originated.

Table 3-46 Malware Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
Malware Event Timestamp	uint32	The malware event generation timestamp.
Event Type ID	uint32	The internal ID of the malware event type.
Event Subtype ID	uint32	The internal ID of the action that led to malware detection.
Detector ID	uint8	The internal ID of the detection technology that detected the malware.
String Block Type	uint32	Initiates a String data block containing the detection name. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Detection Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Detection Name field.
Detection Name	string	The name of the detected or quarantined malware.
String Block Type	uint32	Initiates a String data block containing the username. This value is always `0`.
String Block Length	uint32	The number of bytes included in the User String data block, including eight bytes for the block type and header fields plus the number of bytes in the User field.
User	string	The user of the computer where the Cisco Agent is installed and where the malware event occurred. Note that these users are not tied to user discovery.
String Block Type	uint32	Initiates a String data block containing the file name. This value is always `0`.
String Block Length	uint32	The number of bytes included in the File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Name field.
File Name	string	The name of the detected or quarantined file.
String Block Type	uint32	Initiates a String data block containing the file path. This value is always `0`.
String Block Length	uint32	The number of bytes included in the File Path String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Path field.
File Path	string	The file path, not including the file name, of the detected or quarantined file.
String Block Type	uint32	Initiates a String data block containing the file SHA hash. This value is always `0`.
String Block Length	uint32	The number of bytes included in the File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the File SHA Hash field.
File SHA Hash	string	The rendered string of the SHA-256 hash value of the detected or quarantined file.

Table 3-46 Malware Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
File Size	uint32	The size in bytes of the detected or quarantined file.
File Type	uint32	The file type of the detected or quarantined file. The meaning of this field is transmitted in the metadata with this event. See AMP for Endpoints File Type Metadata, page 3-38 for more information.
File Timestamp	uint32	UNIX timestamp (seconds since 01/01/1970) of the creation of the detected or quarantined file.
String Block Type	uint32	Initiates a String data block containing the parent file name. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Parent File Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File Name field.
Parent File Name	string	The name of the file accessing the detected or quarantined file when detection occurred.
String Block Type	uint32	Initiates a String data block containing the parent file SHA hash. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Parent File SHA Hash String data block, including eight bytes for the block type and header fields plus the number of bytes in the Parent File SHA Hash field.
Parent File SHA Hash	string	The SHA-256 hash value of the parent file accessing the detected or quarantined file when detection occurred.
String Block Type	uint32	Initiates a String data block containing the event description. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Event Description String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Description field.
Event Description	string	The additional event information associated with the event type.
Device ID	uint32	ID for the device that generated the event.
Connection Instance	uint16	Snort instance on the device that generated the event. Used to link the event with a connection or IDS event.
Connection Counter	uint16	Value used to distinguish between connection events that happen during the same second.
Connection Event Timestamp	uint32	Timestamp of the connection event.

Table 3-46 Malware Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
Direction	uint8	Indicates whether the file was uploaded or downloaded. Can have the following values: • 1 — Download • 2 — Upload Currently the value depends on the protocol (for example, if the connection is HTTP it is a download).
Source IP Address	uint8[16]	IPv4 or IPv6 address for the source of the connection.
Destination IP Address	uint8[16]	IPv4 or IPv6 address for the destination of the connection.
Application ID	uint32	ID number that maps to the application using the file transfer.
User ID	uint32	Identification number for the user logged into the destination host, as identified by the system.
Access Control Policy UUID	uint8[16]	Identification number that acts as a unique identifier for the access control policy that triggered the event.
Disposition	uint8	The malware status of the file. Possible values include: • 1 — CLEAN The file is clean and does not contain malware. • 2 — UNKNOWN It is unknown whether the file contains malware. • 3 — MALWARE The file contains malware. • 4 — UNAVAILABLE The software was unable to send a request to the AMP cloud for a disposition, or the AMP cloud services did not respond to the request. • 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user.
Retrospective Disposition	uint8	Disposition of the file if the disposition is updated. If the disposition is not updated, this field contains the same value as the Disposition field. The possible values are the same as the Disposition field.
String Block Type	uint32	Initiates a String data block containing the URI. This value is always 0.
String Block Length	uint32	The number of bytes included in the URI data block, including eight bytes for the block type and header fields plus the number of bytes in the URI field.
URI	string	URI of the connection.
Source Port	uint16	Port number for the source of the connection.
Destination Port	uint16	Port number for the destination of the connection.
Source Country	uint16	Code for the country of the source host.
Destination Country	uint 16	Code for the country of the destination host.

Table 3-46 Malware Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
Web Application ID	uint32	The internal identification number of the detected web application, if applicable.
Client Application ID	uint32	The internal identification number of the detected client application, if applicable.
Action	uint8	The action taken on the file based on the file type. Can have the following values: • 1 — Detect • 2 — Block • 3 — Malware Cloud Lookup • 4 — Malware Block • 5 — Malware Allow List • 6 — Cloud Lookup Timeout • 7 — Custom Detection • 8 — Custom Detection Block • 9 — Archive Block (Depth Exceeded) • 10 — Archive Block (Encrypted) • 11 — Archive Block (Failed to Inspect)
Protocol	uint8	IANA protocol number specified by the user. For example: • 1 — ICMP • 4 — IP • 6 — TCP • 17 — UDP This is currently only TCP.
Threat Score	uint8	A numeric value from 0 to 100 based on the potentially malicious behaviors observed during dynamic analysis.
IOC Number	uint16	ID number of the compromise associated with this event.
Security Context	uint8(16)	ID number for the security context (virtual firewall) that the traffic passed through. Note that the system only populates this field for ASA FirePOWER devices in multi-context mode.
SSL Certificate Fingerprint	uint8[20]	SHA1 hash of the SSL Server certificate.

Table 3-46 Malware Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
SSL Actual Action	uint16	The action performed on the connection based on the SSL Rule. This may differ from the expected action, as the action as specified in the rule may be impossible. Possible values include: • 0 — 'Unknown' • 1 — 'Do Not Decrypt' • 2 — 'Block' • 3 — 'Block With Reset' • 4 — 'Decrypt (Known Key)' • 5 — 'Decrypt (Replace Key)' • 6 — 'Decrypt (Resign)'

Table 3-46 Malware Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
SSL Flow Status	uint16	Status of the SSL Flow. These values describe the reason behind the action taken or the error message seen. Possible values include: • 0 — 'Unknown' • 1 — 'No Match' • 2 — 'Success' • 3 — 'Uncached Session' • 4 — 'Unknown Cipher Suite' • 5 — 'Unsupported Cipher Suite' • 6 — 'Unsupported SSL Version' • 7 — 'SSL Compression Used' • 8 — 'Session Undecryptable in Passive Mode' • 9 — 'Handshake Error' • 10 — 'Decryption Error' • 11 — 'Pending Server Name Category Lookup' • 12 — 'Pending Common Name Category Lookup' • 13 — 'Internal Error' • 14 — 'Network Parameters Unavailable' • 15 — 'Invalid Server Certificate Handle' • 16 — 'Server Certificate Fingerprint Unavailable' • 17 — 'Cannot Cache Subject DN' • 18 — 'Cannot Cache Issuer DN' • 19 — 'Unknown SSL Version' • 20 — 'External Certificate List Unavailable' • 21 — 'External Certificate Fingerprint Unavailable' • 22 — 'Internal Certificate List Invalid' • 23 — 'Internal Certificate List Unavailable' • 24 — 'Internal Certificate Unavailable' • 25 — 'Internal Certificate Fingerprint Unavailable' • 26 — 'Server Certificate Validation Unavailable' • 27 — 'Server Certificate Validation Failure' • 28 — 'Invalid Action'
String Block Type	uint32	Initiates a String data block containing the Archive SHA. This value is always 0.

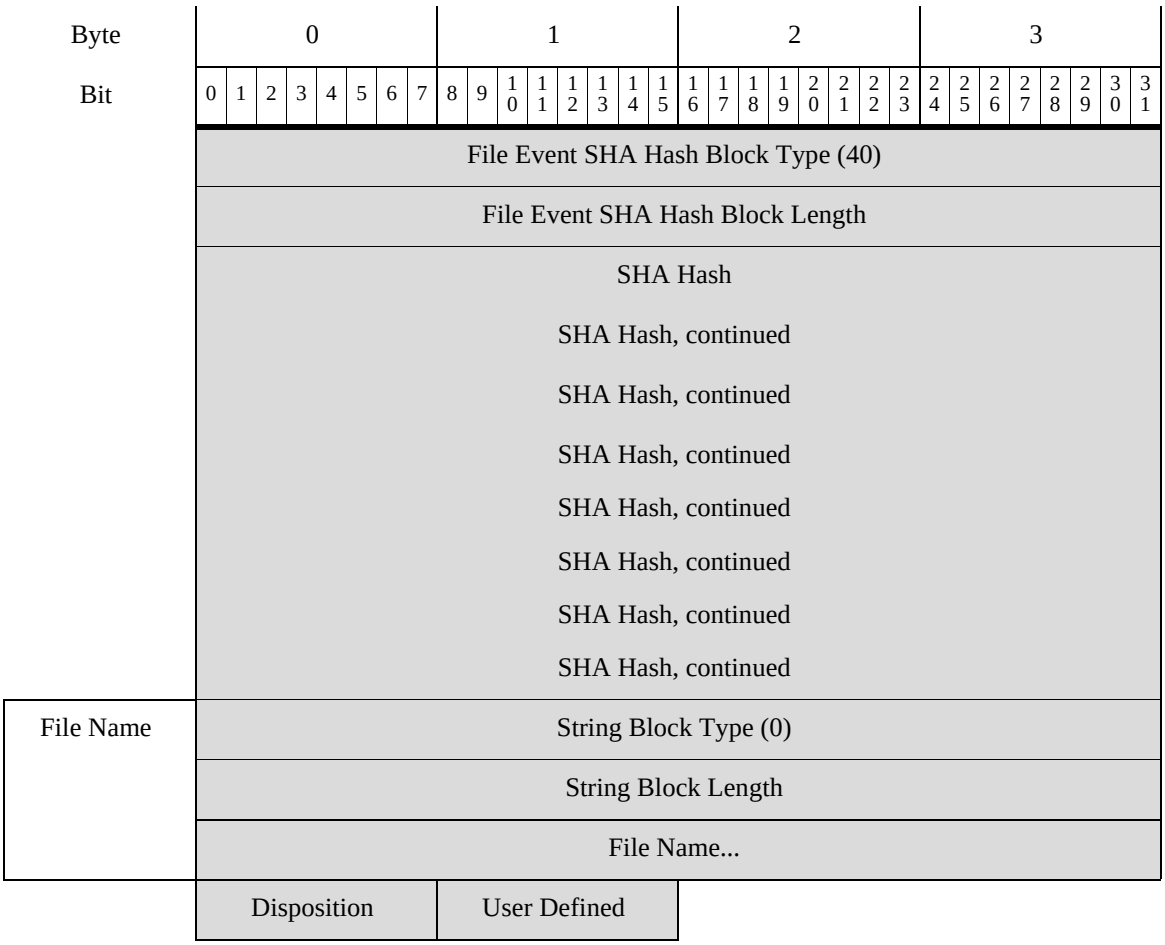
Table 3-46 Malware Event Data Block for 7.0+ Fields (continued)

Field	Data Type	Description
String Block Length	uint32	The number of bytes included in the Archive SHA String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name.
Archive SHA	string	SHA1 hash of the parent archive in which the file is contained.
String Block Type	uint32	Initiates a String data block containing the Archive Name. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Archive Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the intrusion policy name.
Archive Name	string	Name of the parent archive.
Archive Depth	uint8	Number of layers in which the file is nested. For example, if a text file is in a zip archive, this has a value of `1`.
HTTP Response	uint32	Response code of the HTTP Request.
String Block Type	uint32	Initiates a String data block containing the name of the ingress VRF. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Ingress VRF name field.
Ingress VRF Name	string	The virtual router through which traffic entered the network.
String Block Type	uint32	Initiates a String data block containing the name of the Egress VRF. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Egress VRF name field.
Egress VRF Name	string	The name of the virtual router through which traffic exited the network.

File Event SHA Hash for 5.3+

The eStreamer service uses the File Event SHA Hash data block to contain metadata of the mapping of the SHA hash of a file to its filename. The block type is 40 in the series 2 list of data blocks. It can be requested if file log events have been requested in the extended requests—event code `111`—and either bit 20 is set or metadata is requested with an event version of `5` and an event code of `21`.

The following diagram shows the structure of a file event hash data block:



The following table describes the fields in the file event SHA hash data block.

Table 3-47 File Event SHA Hash Data Block Fields

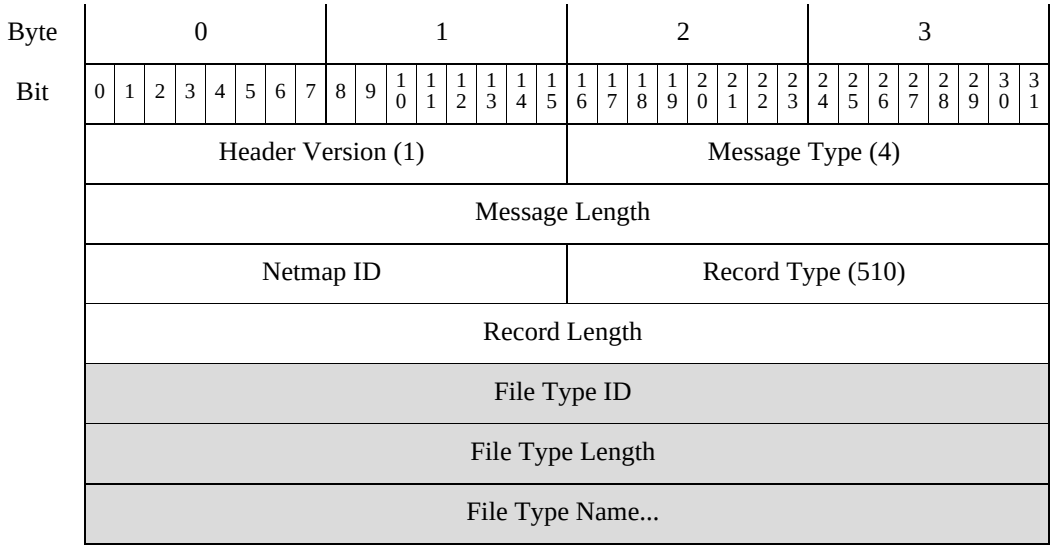
Field	Data Type	Description
File Event SHA Hash Block Type	uint32	Initiates a File Event SHA Hash block. This value is always `40`.
File Event SHA Hash Block Length	uint32	Total number of bytes in the File Event SHA Hash block, including eight bytes for the File Event SHA Hash block type and length fields, plus the number of bytes of data that follows.
SHA Hash	uint8[32]	The SHA-256 hash of the file in binary format.
String Block Type	uint32	Initiates a String data block containing the descriptive name associated with the file. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Name field.
File Name or Disposition	string	The descriptive name or disposition of the file. If the file is clean, this value is `Clean`. If the file's disposition is unknown, the value is `Neutral`. If the file contains malware, the file name is given.

Table 3-47 File Event SHA Hash Data Block Fields (continued)

Field	Data Type	Description
Disposition	uint8	The malware status of the file. Possible values include: • 1 — CLEAN The file is clean and does not contain malware. • 2 — UNKNOWN It is unknown whether the file contains malware. • 3 — MALWARE The file contains malware. • 4 — UNAVAILABLE The software was unable to send a request to the AMP cloud for a disposition, or the AMP cloud services did not respond to the request. • 5 — CUSTOM SIGNATURE The file matches a user-defined hash, and is treated in a fashion designated by the user
User Defined	uint8	Indicated how the file name was provided: • 0 — Defined by AMP • 1 — User defined

File Type ID Metadata for 5.3+

The eStreamer service transmits metadata containing file type information for an event with a file type id, the format of which is shown below. This record maps a file type id to a file type name. File type ID information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of 510, indicating a file type id record.



The following table describes the fields in the File Type ID record.

Table 3-48 File Type ID Record Fields

Field	Data Type	Description
File Type ID	uint32	File Type ID number. This field is the unique key for this record.
File Type Length	uint32	The number of bytes included in the file type name.
File Type Name	string	The descriptive name for the file type.

Rule Documentation Data Block for 5.2+

The eStreamer service uses the Rule Documentation data block to contain information about rules used to generate alerts. The block type is 27 in the series 2 set of data blocks. It can be requested with a host request message of type 10. See Host Request Message Format, page 2-27 for more information.

The following diagram shows the structure of a rule documentation data block:

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Rule Documentation Block Type (27)																															
	Rule Documentation Block Length																															
	Signature ID																															
	Generator ID																															
	Revision																															
Summary	String Block Type (0)																															
	String Block Length																															
	Summary...																															
Impact	String Block Type (0)																															
	String Block Length																															
	Impact...																															
Detailed Info	String Block Type (0)																															
	String Block Length																															
	Detailed Information																															

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Affected Systems	String Block Type (0)																															
	String Block Length																															
	Affected Systems...																															
Attack Scenarios	String Block Type (0)																															
	String Block Length																															
	Attack Scenarios...																															
Ease of Attack	String Block Type (0)																															
	String Block Length																															
	Ease of Attack...																															
False Positives	String Block Type (0)																															
	String Block Length																															
	False Positives...																															
False Negatives	String Block Type (0)																															
	String Block Length																															
	False Negatives...																															
Corrective Action	String Block Type (0)																															
	String Block Length																															
	Corrective Action...																															
Contributors	String Block Type (0)																															
	String Block Length																															
	Contributors...																															
Additional References	String Block Type (0)																															
	String Block Length																															
	Additional References...																															

The following table describes the fields in the rule documentation data block.

Table 3-49 Rule Documentation Data Block Fields

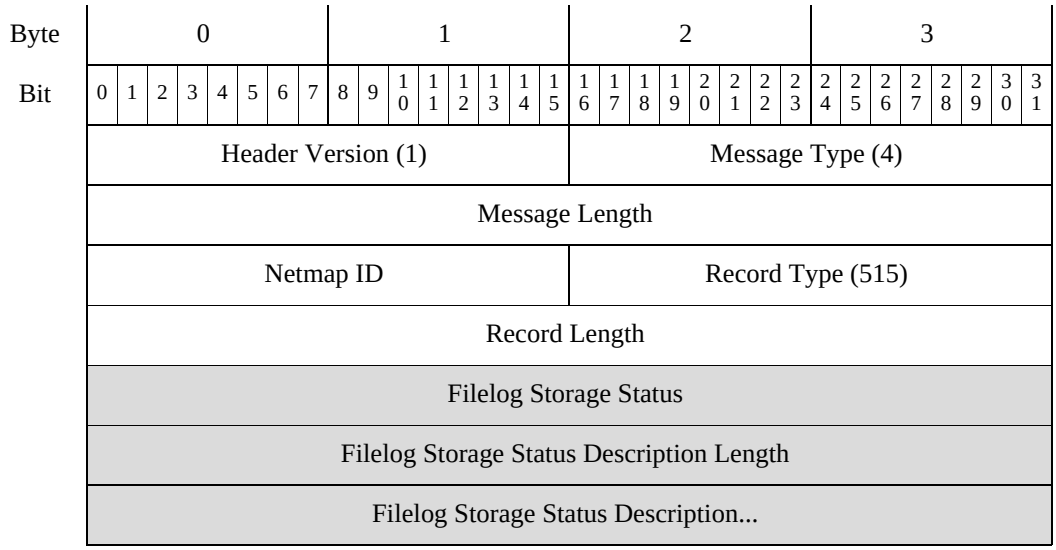
Field	Data Type	Description
Rule Documentation Data Block Type	uint32	Initiates a Rule Documentation data block. This value is always 27.
Rule Documentation Data Block Length	uint32	Total number of bytes in the Rule Documentation data block, including eight bytes for the Rule Documentation data block type and length fields, plus the number of bytes of data that follows.
Rule ID (Signature ID)	uint32	Rule identification number that corresponds with the event.
Generator ID	uint32	Identification number of the Secure Firewall System preprocessor that generated the event.
Rule Revision	uint32	Rule revision number.
String Block Type	uint32	Initiates a String data block containing the summary associated with the rule. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Summary field.
Summary	string	Explanation of the threat or vulnerability.
String Block Type	uint32	Initiates a String data block containing the impact associated with the rule. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Impact field.
Impact	string	How a compromise that uses this vulnerability may impact various systems.
String Block Type	uint32	Initiates a String data block containing the detailed information associated with the rule. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Detailed Information field.
Detailed Information	string	Information regarding the underlying vulnerability, what the rule actually looks for, and what systems are affected.
String Block Type	uint32	Initiates a String data block containing the list of affected systems associated with the rule. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Affected Systems field.
Affected Systems	string	Systems affected by the vulnerability.
String Block Type	uint32	Initiates a String data block containing the possible attack scenarios associated with the rule. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Attack Scenarios field.

Table 3-49 Rule Documentation Data Block Fields (continued)

Field	Data Type	Description
Attack Scenarios	string	Examples of possible attacks.
String Block Type	uint32	Initiates a String data block containing the ease of attack associated with the rule. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Ease of Attack field.
Ease of Attack	string	Whether the attack is considered simple, medium, hard, or difficult, and whether or not is can be performed using a script.
String Block Type	uint32	Initiates a String data block containing the possible false positives associated with the rule. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the False Positives field.
False Positives	string	Examples that may result in a false positive. The default value is `None Known`.
String Block Type	uint32	Initiates a String data block containing the possible false negatives associated with the rule. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the False Negatives field.
False Negatives	string	Examples that may result in a false negative. The default value is `None Known`.
String Block Type	uint32	Initiates a String data block containing the corrective action associated with the rule. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Corrective Action field.
Corrective Action	string	Information regarding patches, upgrades, or other means to remove or mitigate the vulnerability.
String Block Type	uint32	Initiates a String data block containing the contributors for the rule. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Contributors field.
Contributors	string	Contact information for the author of the rule and other relevant documentation.
String Block Type	uint32	Initiates a String data block containing the additional references associated with the rule. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Additional References field.
Additional References	string	Additional information and references.

Filelog Storage Metadata for 6.0+

The eStreamer service transmits metadata containing filelog storage information. Note that the Record Type field, which appears after the Message Length field, has a value of `515`, indicating a Filelog Storage Metadata record.



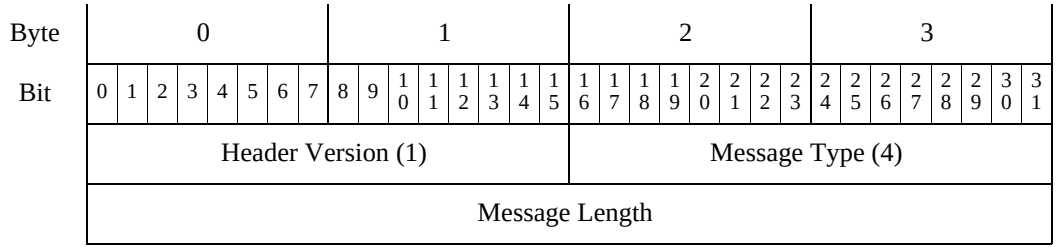
The following table describes the fields in the Filelog Storage Metadata record.

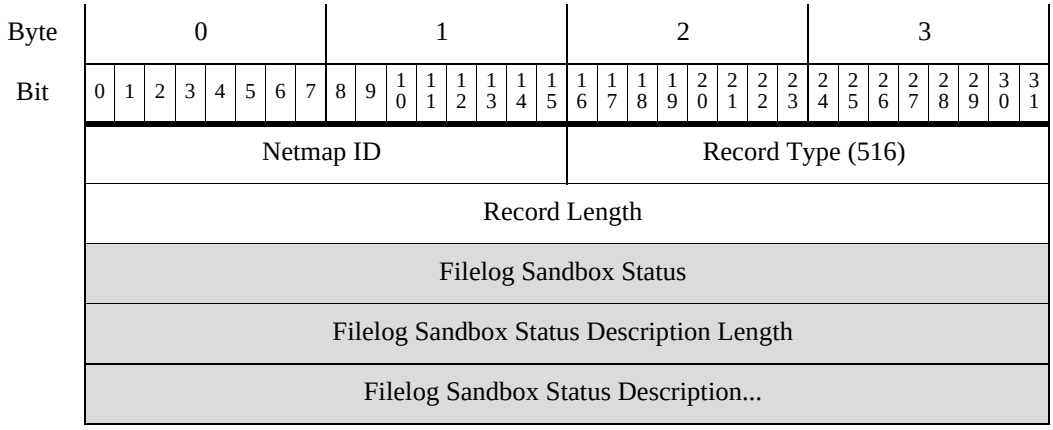
Table 3-50 Filelog Storage Metadata Record Fields

Field	Data Type	Description
Filelog Storage Status	uint32	Number denoting the filelog storage status. This field is the unique key for this record.
Filelog Storage Status Description Length	uint32	The number of bytes included in the Filelog Storage Status Description.
Filelog Storage Status Description	string	The descriptive name for the filelog storage status.

Filelog Sandbox Metadata for 6.0+

The eStreamer service transmits metadata containing filelog sandbox information. Note that the Record Type field, which appears after the Message Length field, has a value of `516`, indicating a Filelog Sandbox Metadata record.





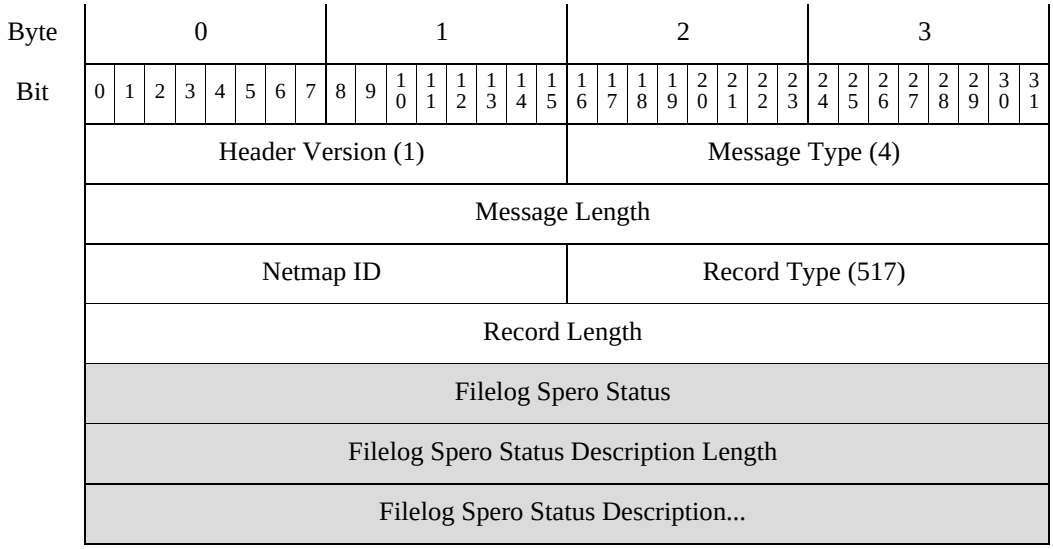
The following table describes the fields in the Filelog Sandbox Metadata record.

Table 3-51 Filelog Sandbox Metadata Record Fields

Field	Data Type	Description
Filelog Sandbox Status	uint32	Number denoting the filelog sandbox status. This field is the unique key for this record.
Filelog Sandbox Status Description Length	uint32	The number of bytes included in the Filelog Sandbox Status Description.
Filelog Sandbox Status Description	string	The descriptive name for the filelog sandbox status.

Filelog Spero Metadata for 6.0+

The eStreamer service transmits metadata containing filelog spero information. Note that the Record Type field, which appears after the Message Length field, has a value of `517`, indicating a filelog spero metadata record.



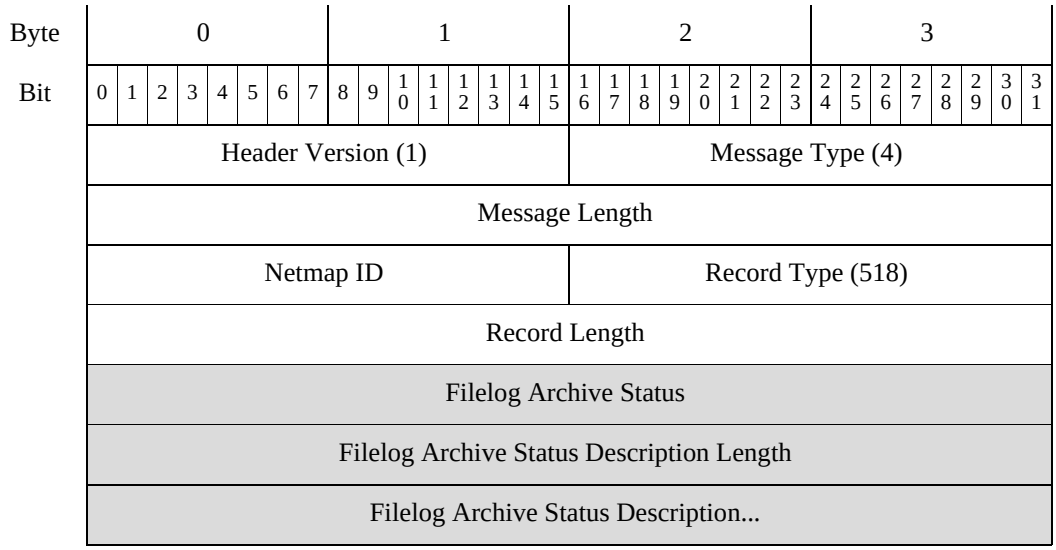
The following table describes the fields in the Filelog Spero Metadata record.

Table 3-52 Filelog Spero Metadata Record Fields

Field	Data Type	Description
Filelog Spero Status	uint32	Number denoting the filelog spero status. This field is the unique key for this record.
Filelog Spero Status Description Length	uint32	The number of bytes included in the Filelog Spero Status Description.
Filelog Spero Status Description	string	The descriptive name for the filelog spero status.

Filelog Archive Metadata for 6.0+

The eStreamer service transmits metadata containing filelog archive information. Note that the Record Type field, which appears after the Message Length field, has a value of `518`, indicating a Filelog Archive Metadata record.



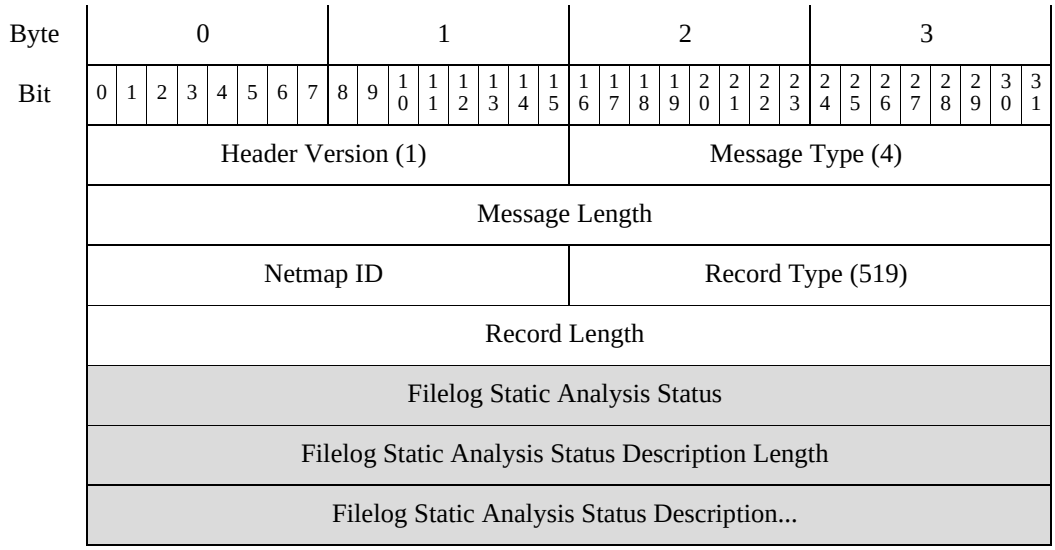
The following table describes the fields in the Filelog Archive Metadata record.

Table 3-53 Filelog Archive Metadata Record Fields

Field	Data Type	Description
Filelog Archive Status	uint32	Number denoting the filelog archive status. This field is the unique key for this record.
Filelog Archive Status Description Length	uint32	The number of bytes included in the Filelog Archive Status Description.
Filelog Archive Status Description	string	The descriptive name for the filelog archive status.

Filelog Static Analysis Metadata for 6.0+

The eStreamer service transmits metadata containing filelog static analysis information. Note that the Record Type field, which appears after the Message Length field, has a value of `519`, indicating a Filelog Static Analysis Metadata record.



The following table describes the fields in the Filelog Static Analysis Metadata record.

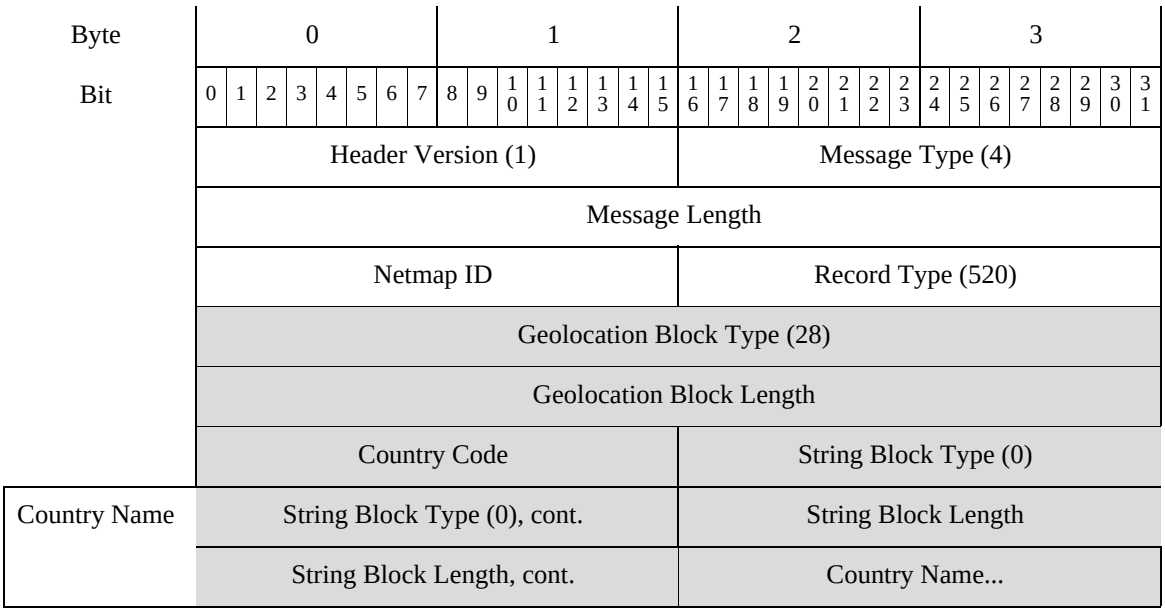
Table 3-54 Filelog Static Analysis Metadata Record Fields

Field	Data Type	Description
Filelog Static Analysis Status	uint32	Number denoting the filelog static analysis status. This field is the unique key for this record.
Filelog Static Analysis Status Description Length	uint32	The number of bytes included in the Filelog Static Analysis Status Description.
Filelog Static Analysis Status Description	string	The descriptive name for the filelog static analysis status.

Geolocation Data Block for 5.2+

This is a data block that contains the mapping of a country code to a country name. The record type is 520, and a block type of 28 in series 2. It is exposed as metadata for any event that has geolocation information. If metadata is requested and there is a value for the country code(s) in the event, then this block is returned along with other metadata.

The following diagram shows the structure of a geolocation data block:



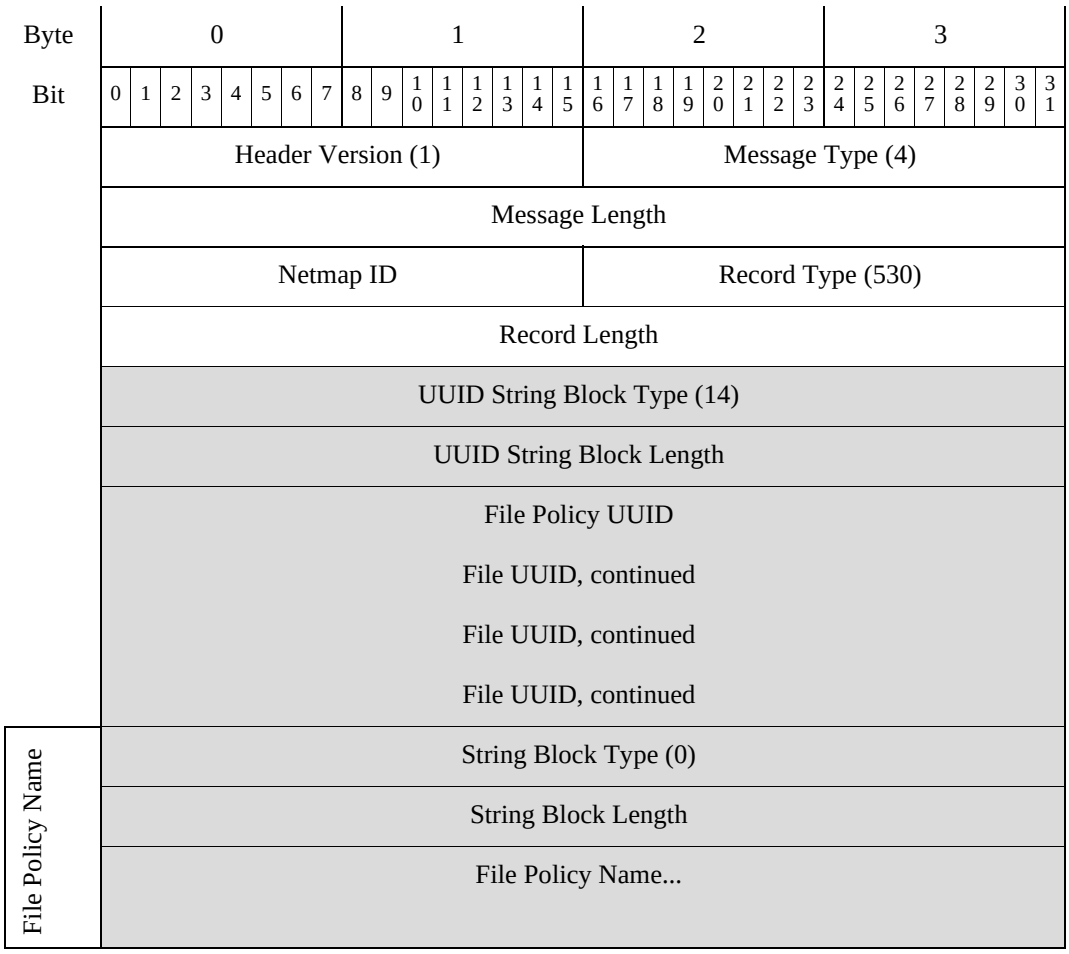
The following table describes the fields in the Geolocation data block.

Table 3-55 Geolocation Data Block Fields

Field	Data Type	Description
Geolocation Data Block Type	uint32	Initiates a Geolocation data block. This value is always `28`.
Geolocation Data Block Length	uint32	Total number of bytes in the Geolocation data block, including eight bytes for the Geolocation data block type and length fields, plus the number of bytes of data that follows.
Country Code	uint16	The country code.
String Block Type	uint32	Initiates a String data block containing the country name associated with the country code. This value is always `0`.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Country Name field.
Country Name	string	The name of the country associated with the country code.

File Policy Name for 6.0+

The eStreamer service transmits metadata containing File Policy Name information, the format of which is shown below. (File Policy Name information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `530`, indicating a File Policy Name record.



The following table describes the fields in the File Policy Name record.

Table 3-56 File Policy Name Fields

Field	Data Type	Description
UUID String Data Block Type	uint32	Initiates a UUID String data block. This value is always `14`.
UUID String Data Block Length	uint32	Total number of bytes in the UUID String data block, including eight bytes for the UUID String data block type and length fields, plus the number of bytes of data that follows.
File Policy UUID	uint8[16]	The UUID of the File Policy. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the File Policy. This value is always `0`.

Table 3-56 File Policy Name Fields (continued)

Field	Data Type	Description
String Block Length	uint32	The number of bytes included in the SSL Policy Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the File Policy name.
File Policy Name	string	The name of the File Policy.

SSL Policy Name

The eStreamer service transmits metadata containing SSL Policy Name information, the format of which is shown below. (SSL Policy Name information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `600`, indicating a SSL Policy Name record.

	Byte	0								1								2								3							
	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
		Header Version (1)																Message Type (4)															
		Message Length																															
		Netmap ID																Record Type (600)															
		Record Length																															
		UUID String Block Type (14)																															
		UUID String Block Length																															
		SSL Policy UUID SSL Policy UUID, continued SSL Policy UUID, continued SSL Policy UUID, continued																															
SSL Policy Name		String Block Type (0)																															
		String Block Length																															
		SSL Policy Name...																															

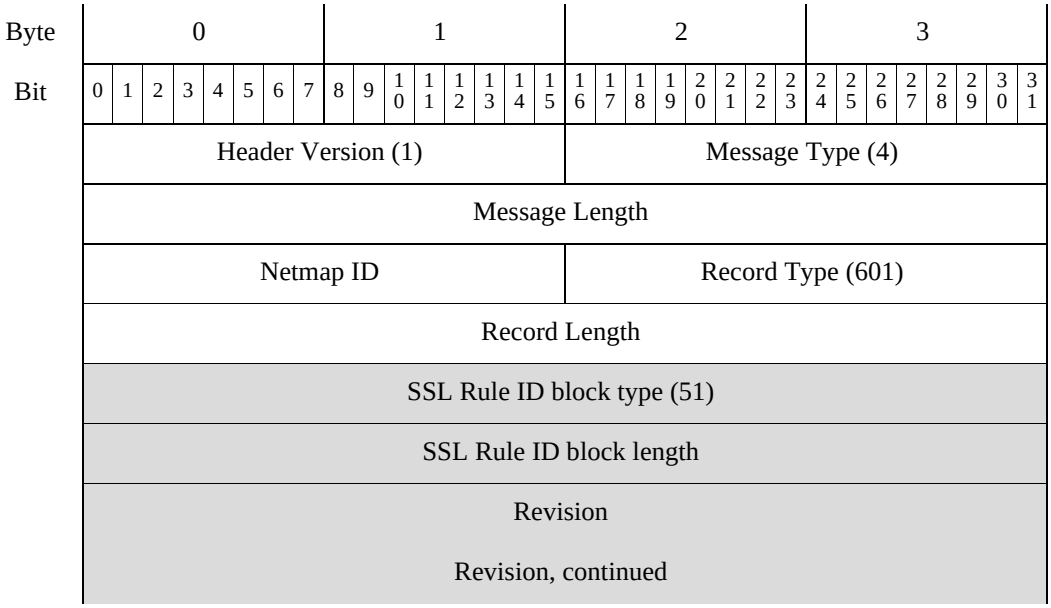
The following table describes the fields in the SSL Policy Name record.

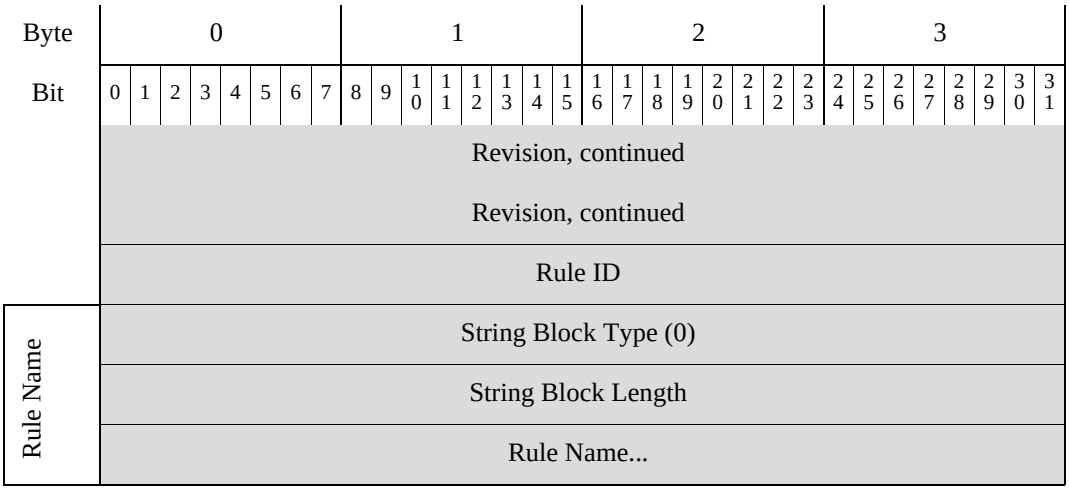
Table 3-57 SSL Policy Name Record Fields

Field	Data Type	Description
UUID String Data Block Type	uint32	Initiates a UUID String data block. This value is always `14`.
UUID String Data Block Length	uint32	Total number of bytes in the UUID String data block, including eight bytes for the UUID String data block type and length fields, plus the number of bytes of data that follows.
SSL Policy UUID	uint8[16]	The UUID of the SSL Policy. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the SSL Policy. This value is always `0`.
String Block Length	uint32	The number of bytes included in the SSL Policy Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SSL Policy name.
SSL Policy Name	string	The name of the SSL Policy.

SSL Rule ID

The eStreamer service transmits metadata containing SSL Rule ID information, the format of which is shown below. (SSL Rule ID information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `601`, indicating a SSL Rule ID record.





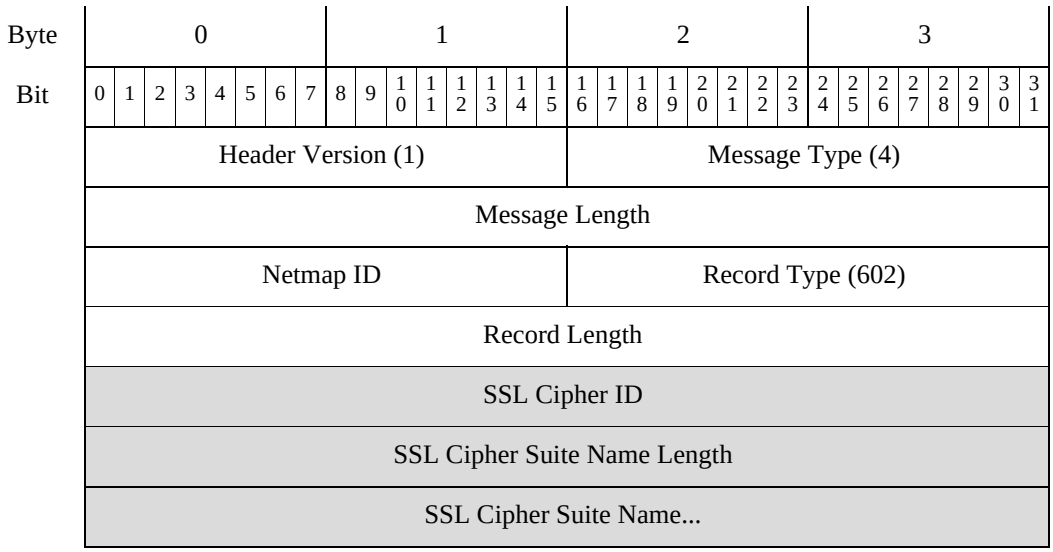
The following table describes the fields in the SSL Rule ID record.

Table 3-58 SSL Policy Name Record Fields

Field	Data Type	Description
SSL Rule ID Block Type	uint32	Block type of the SSL Rule ID data block. This value is always `51`.
SSL Rule ID Block Length	uint32	The number of bytes in the SSL Rule ID data block, including 8 bytes for the block type and header fields plus the number of bytes in the SSL Rule ID block.
Revision	uint8[16]	The UUID of the SSL Rule Revision. This field, combined with the Rule ID, make up the unique key for this record.
Rule ID	uint32	ID number of the SSL Rule. This field, combined with the Revision, make up the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the SSL Rule. This value is always `0`.
String Block Length	uint32	The number of bytes included in the SSL Rule Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the SSL Rule Name.
SSL Rule Name	string	The name of the SSL Rule.

SSL Cipher Suite

The eStreamer service transmits metadata containing SSL Cipher Suite information for an event with a SSL Cipher id, the format of which is shown below. This record maps a SSL Cipher id to a SSL Cipher Suite name. SSL Cipher Suite information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `602`, indicating a SSL Cipher Suite record.



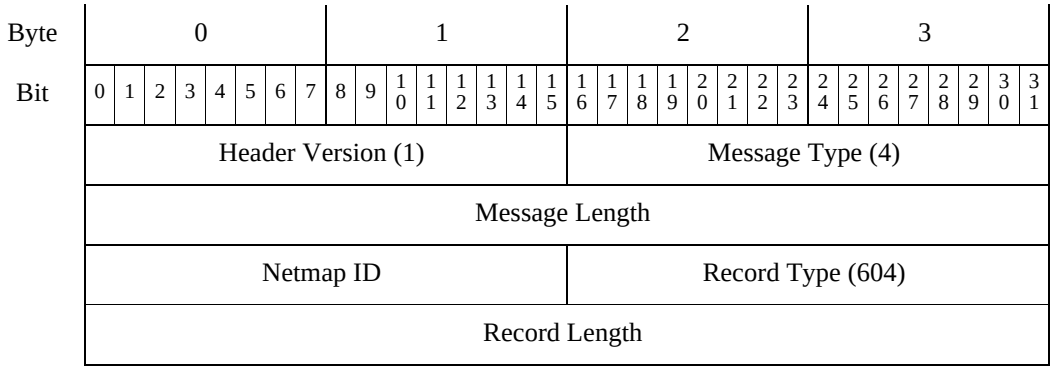
The following table describes the fields in the SSL Cipher Suite record.

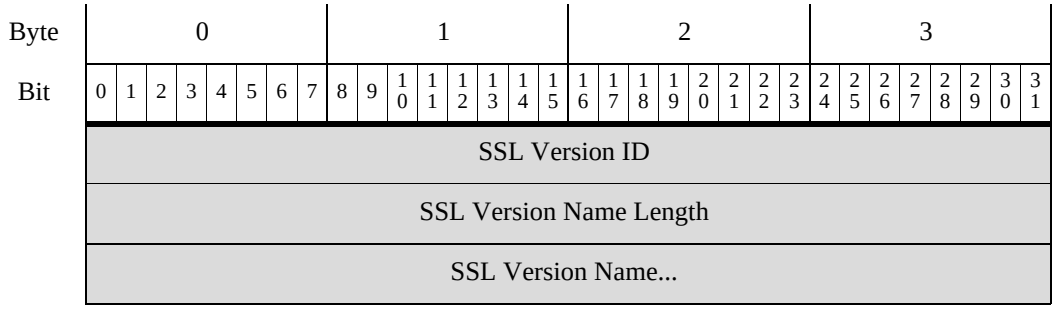
Table 3-59 SSL Cipher Suite Fields

Field	Data Type	Description
SSL Cipher ID	uint32	SSL Cipher ID number. This field is the unique key for this record.
SSL Cipher Suite Name Length	uint32	The number of bytes included in the SSL cipher suite name.
SSL Cipher Suite Name	string	The descriptive name for the SSL Cipher Suite.

SSL Version

The eStreamer service transmits metadata containing SSL Version information for an event with a SSL Version, the format of which is shown below. This record maps a SSL Version ID to a SSL Version name. SSL Cipher Suite information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `604`, indicating a SSL Version record.





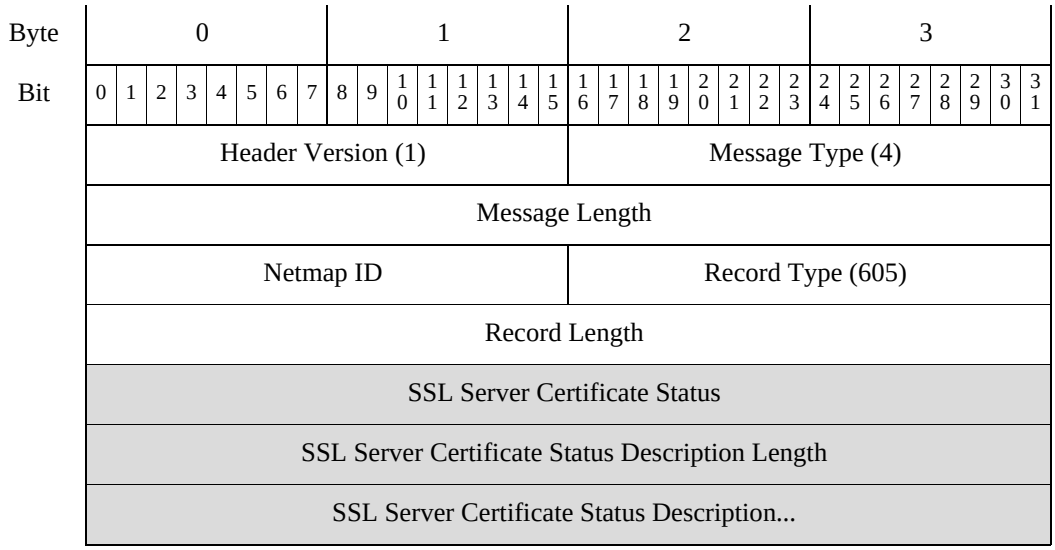
The following table describes the fields in the SSL Version record.

Table 3-60 SSL Version Fields

Field	Data Type	Description
SSL Version ID	uint32	SSL Version ID number. This field is the unique key for this record.
SSL Version Name	uint32	The number of bytes included in the SSL Version Name.
SSL Cipher Suite Name	string	The descriptive name for the SSL Version.

SSL Server Certificate Status

The eStreamer service transmits metadata containing SSL Server Certificate Status information, the format of which is shown below. (SSL Server Certificate Status information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `605`, indicating a SSL Server Certificate Status record.



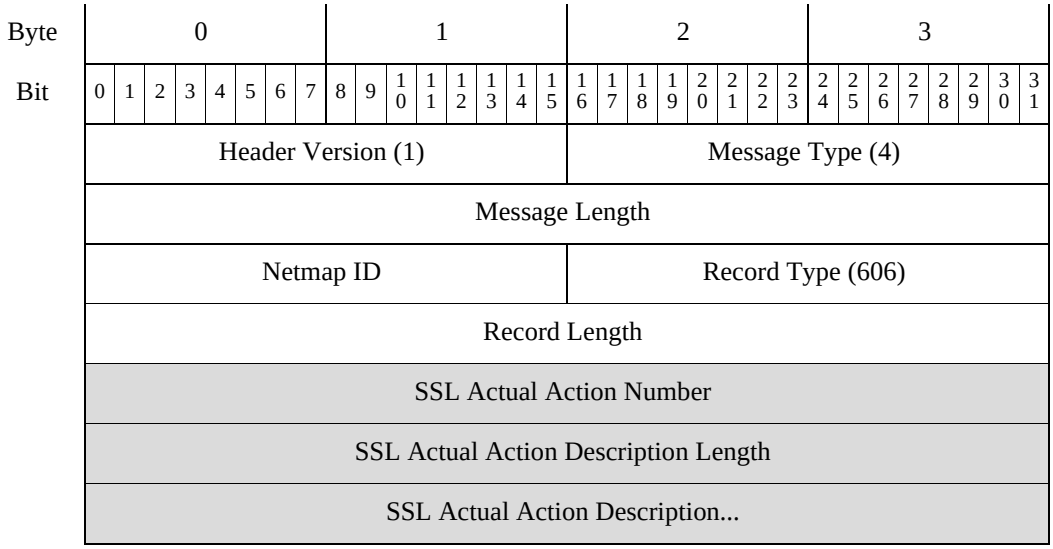
The following table describes the fields in the SSL Server Certificate Status record.

Table 3-61 SSL Server Certificate Status Record Fields

Field	Data Type	Description
SSL Server Certificate Status	uint32	The SSL Server Certificate Status Number. This field is the unique key for this record.
SSL Server Certificate Status Description Lenth	uint32	The number of bytes included in the SSL Server Certificate Status Description.
SSL Server Certificate Status Description	string	The description of the SSL Server Certificate Status.

SSL Actual Action

The eStreamer service transmits metadata containing SSL Actual Action information, the format of which is shown below. (SSL Actual Action information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `606`, indicating a SSL Actual Action record.



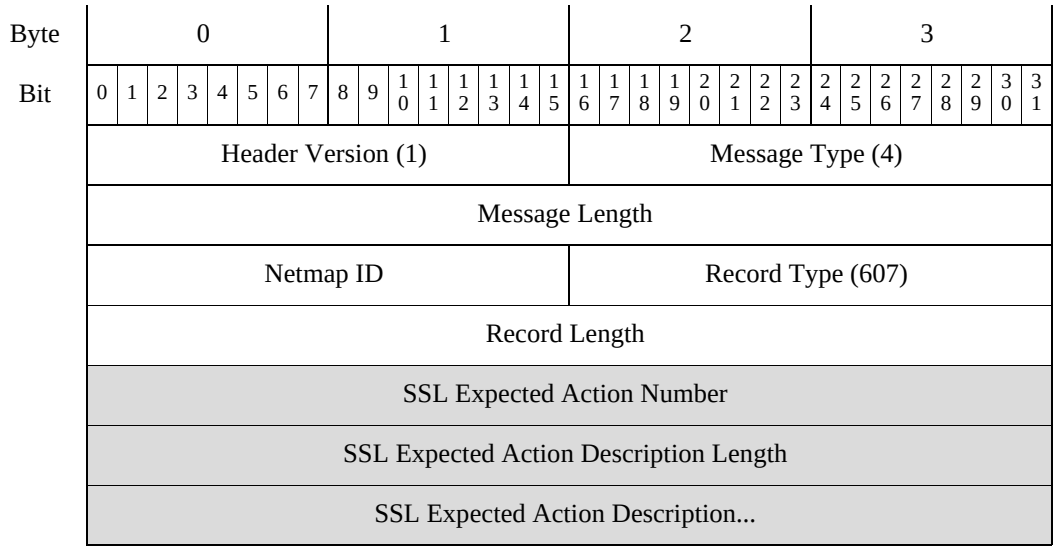
The following table describes the fields in the SSL Actual Action record.

Table 3-62 SSL Actual Action Fields

Field	Data Type	Description
SSL Actual Action Number	uint32	The number designating the SSL Actual Action. This field is the unique key for this record.
SSL Actual Action Description Length	uint32	The number of bytes included in the SSL Actual Action Description.
SSL Actual Action Description	string	The description of the SSL Actual Action.

SSL Expected Action

The eStreamer service transmits metadata containing SSL Expected Action information, the format of which is shown below. (SSL Expected Action information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `607`, indicating a SSL Expected Action record.



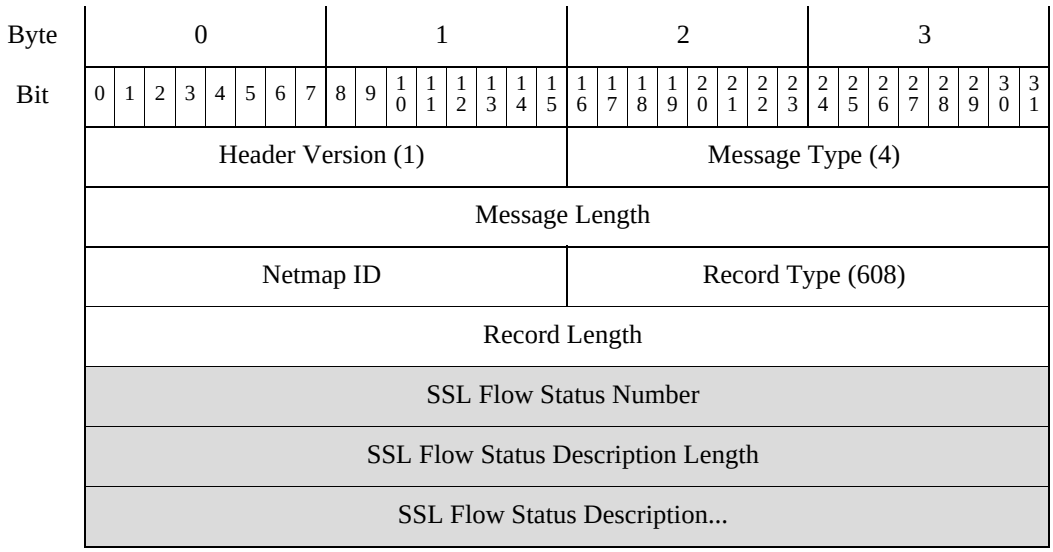
The following table describes the fields in the SSL Expected Action record.

Table 3-63 SSL Actual Action Fields

Field	Data Type	Description
SSL Expected Action Number	uint32	The number designating the SSL Expected Action. This field is the unique key for this record.
SSL Expected Action Description Length	uint32	The number of bytes included in the SSL Expected Action Description.
SSL Expected Action Description	string	The description of the SSL Expected Action.

SSL Flow Status

The eStreamer service transmits metadata containing SSL Flow Status information, the format of which is shown below. (SSL Flow Status information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `608`, indicating a SSL Flow Status record.



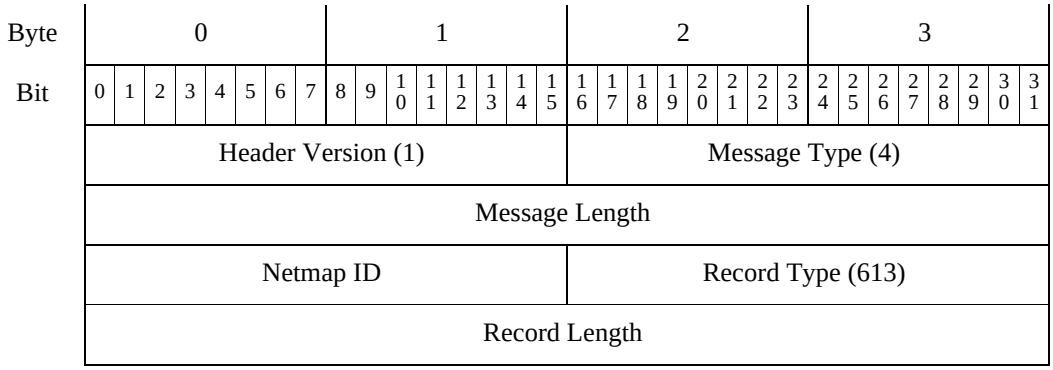
The following table describes the fields in the SSL Flow Status record.

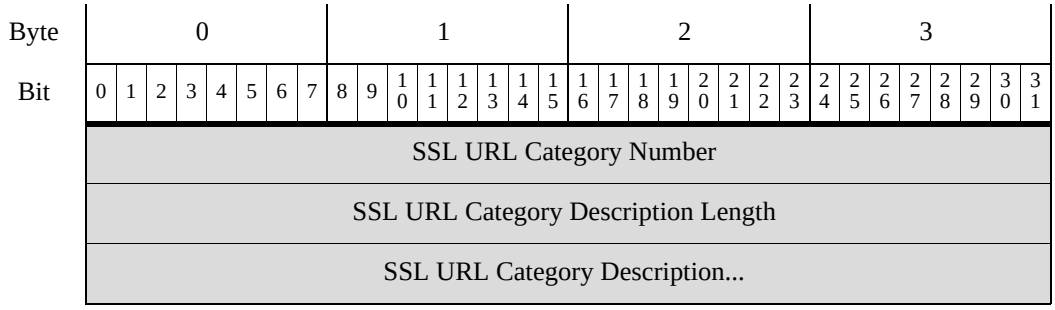
Table 3-64 SSL Flow Status Fields

Field	Data Type	Description
SSL Flow Status Number	uint32	The number designating the SSL Flow Status. This field is the unique key for this record.
SSL Flow Status Description Length	uint32	The number of bytes included in the SSL Flow Status Description.
SSL Flow Status Description	string	The description of the SSL Flow Status.

SSL URL Category

The eStreamer service transmits metadata containing SSL URL Category information, the format of which is shown below. (SSL URL Category information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of `613`, indicating a SSL URL Category record.





The following table describes the fields in the SSL URL Category record.

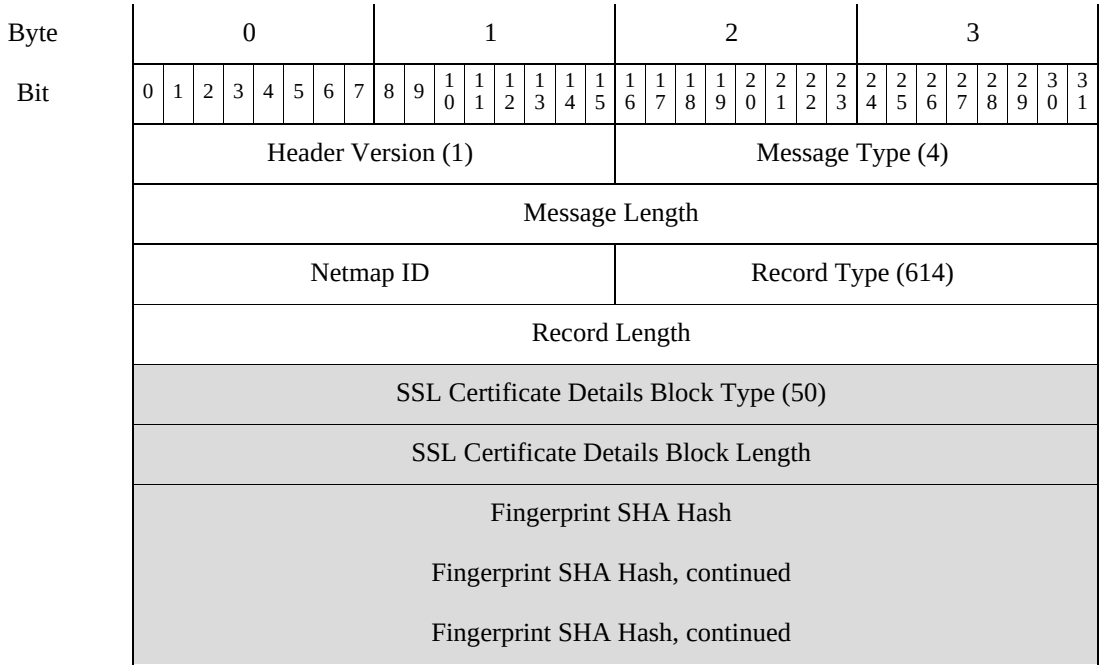
Table 3-65 SSL URL Category Fields

Field	Data Type	Description
SSL URL Category Number	uint32	The number designating the SSL URL Category. This field is the unique key for this record.
SSL URL Category Description Length	uint32	The number of bytes included in the SSL Server URL Category Description.
SSL URL Category Description	string	The description of the SSL URL Category.

SSL Certificate Details Data Block for 5.4+

This is a data block that provides detailed information regarding an SSL certificate. The record type is 614, with a block type of 50 in series 2. It is exposed as metadata for any event that has SSL information. These include malware events, file events, intrusion events, connection events, and correlation events.

The following diagram shows the structure of an SSL Certificate Details data block:



Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Fingerprint SHA Hash, continued																															
	Fingerprint SHA Hash, continued																															
	Public Key SHA Hash																															
	Public Key SHA Hash, continued																															
	Public Key SHA Hash, continued																															
	Public Key SHA Hash, continued																															
	Public Key SHA Hash, continued																															
	Serial Number																															
	Serial Number, continued																															
	Serial Number, continued																															
	Serial Number, continued																															
	Serial Number, continued																															
	Serial Number Length																															
Subject Common Name	String Block Type (0)																															
	String Block Length																															
	Subject Common Name...																															
Subject Organization	String Block Type (0)																															
	String Block Length																															
	Subject Organization...																															
Subject Organizational Unit	String Block Type (0)																															
	String Block Length																															
	Subject Organizational Unit....																															
Subject Country	String Block Type (0)																															
	String Block Length																															
	Subject Country...																															

Byte	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Issuer Common Name	String Block Type (0)																															
	String Block Length																															
	Issuer Common Name...																															
Issuer Organization	String Block Type (0)																															
	String Block Length																															
	Issuer Organization...																															
Issuer Organizational Unit	String Block Type (0)																															
	String Block Length																															
	Issuer Organizational Unit...																															
Issuer Country	String Block Type (0)																															
	String Block Length																															
	Issuer Country...																															
	Valid Start Date																															
	Valid End Date																															

The following table describes the fields in the SSL Certificate Details data block.

Table 3-66 SSL Certificate Details Data Block Fields

Field	Data Type	Description
SSL Certificate Details Data Block Type	uint32	Initiates an SSL Certificate Details data block. This value is always `50`.
SSL Certificate Details Data Block Length	uint32	Total number of bytes in the SSL Certificate Details data block, including eight bytes for the SSL Certificate Details data block type and length fields, plus the number of bytes of data that follows.
Fingerprint SHA Hash	uint8[20]	SHA1 hash of the SSL Server certificate.
Public Key SHA Hash	uint8[20]	The SHA hash value used to authenticate the public key contained within the certificate.
Serial Number	uint8[20]	The serial number assigned by the issuing CA. While this number cannot exceed 20 bytes in length, it can be less than 20 bytes as specified in the Serial Number Length field.

Table 3-66 SSL Certificate Details Data Block Fields (continued)

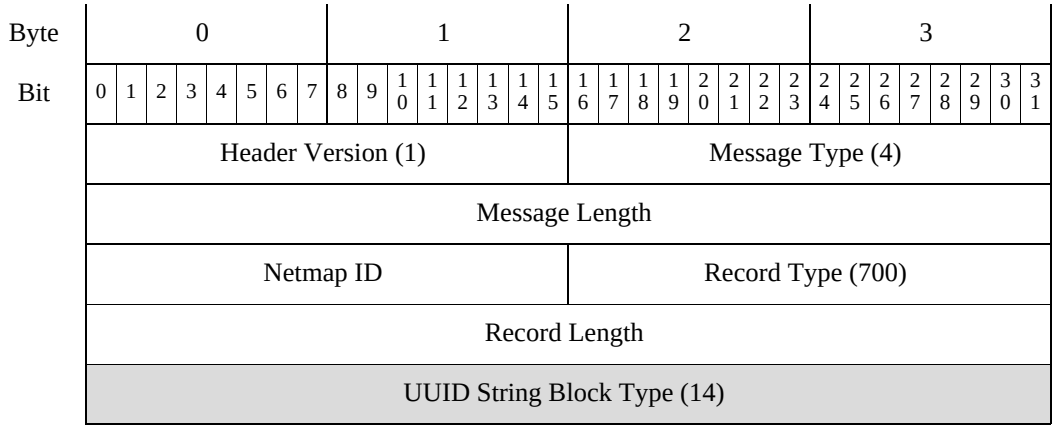
Field	Data Type	Description
Serial Number Length	uint32	The length of the serial number in bytes.
String Block Type	uint32	Initiates a String data block containing the category associated with the compromise. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Category field.
Subject Common Name	string	Subject Common name from the SSL Certificate This is typically the host and domain name of the certificate subject, but may contain other information.
String Block Type	uint32	Initiates a String data block containing the event type associated with the compromise. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Type field.
Subject Organization	string	The organization of the certificate subject.
String Block Type	uint32	Initiates a String data block containing the event type associated with the compromise. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Type field.
Subject Organizational Unit	string	The organizational unit of the certificate subject.
String Block Type	uint32	Initiates a String data block containing the event type associated with the compromise. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Type field.
Subject Country	string	The country of the certificate subject.
String Block Type	uint32	Initiates a String data block containing the category associated with the compromise. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Category field.
Issuer Common Name	string	Issuer Common name from the SSL Certificate This is typically the host and domain name of the certificate issuer, but may contain other information.
String Block Type	uint32	Initiates a String data block containing the event type associated with the compromise. This value is always 0.

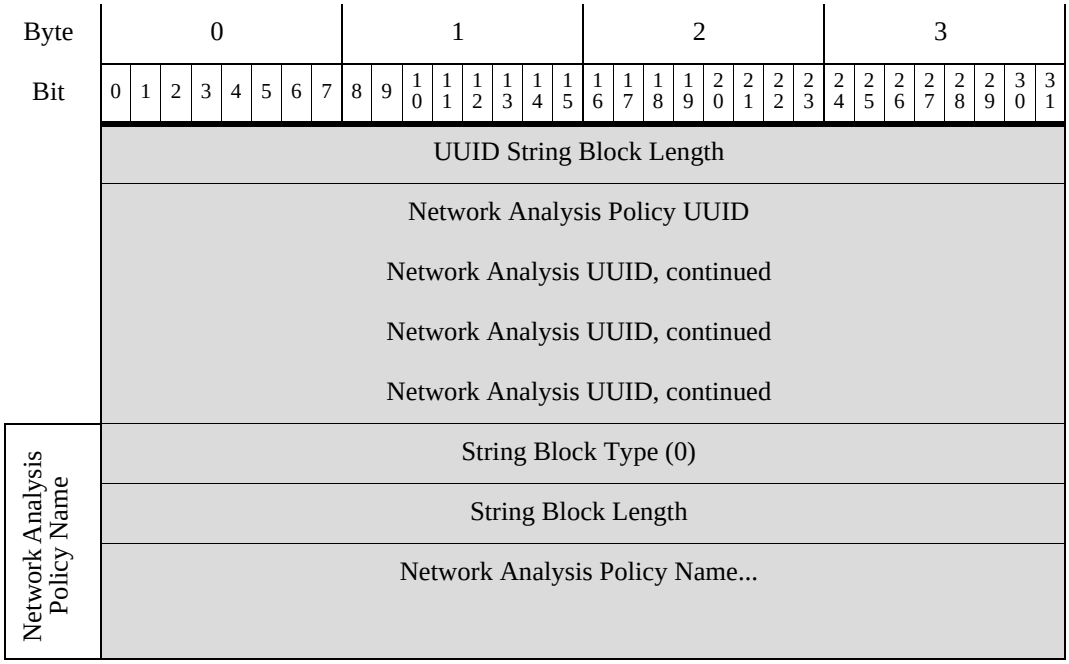
Table 3-66 SSL Certificate Details Data Block Fields (continued)

Field	Data Type	Description
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Type field.
Issuer Organization	string	The organization of the certificate issuer.
String Block Type	uint32	Initiates a String data block containing the event type associated with the compromise. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Type field.
Issuer Organizational Unit	string	The organizational unit of the certificate issuer.
String Block Type	uint32	Initiates a String data block containing the event type associated with the compromise. This value is always 0.
String Block Length	uint32	The number of bytes included in the name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Event Type field.
Issuer Country	string	The country of the certificate issuer.
Valid Start Date	uint32	The Unix timestamp when the certificate was issued.
Valid End Date	uint32	The Unix timestamp on which the certificate ceases to be valid.

Network Analysis Policy Name Record

The eStreamer service transmits metadata containing Network Analysis Policy Name information, the format of which is shown below. (Network Analysis Policy Name information is sent when one of the metadata flags—bits 1, 14, 15, or 20 in the Request Flags field of a request message—is set. See Request Flags, page 2-13.) Note that the Record Type field, which appears after the Message Length field, has a value of 700, indicating a Network Analysis Policy Name record.





The following table describes the fields in the Network Analysis Policy Name record.

Table 3-67 Network Analysis Policy Name Record Fields

Field	Data Type	Description
UUID String Data Block Type	uint32	Initiates a UUID String data block. This value is always `14`.
UUID String Data Block Length	uint32	Total number of bytes in the UUID String data block, including eight bytes for the UUID String data block type and length fields, plus the number of bytes of data that follows.
Network Analysis Policy UUID	uint8[16]	The UUID of the Network Analysis Policy. This field is the unique key for this record.
String Block Type	uint32	Initiates a String data block containing the name of the Network Analysis Policy. This value is always `0`.
String Block Length	uint32	The number of bytes included in the Network Analysis Policy Name String data block, including eight bytes for the block type and header fields plus the number of bytes in the Network Analysis Policy name.
Network Analysis Policy Name	string	The name of the Network Analysis Policy.